NTNU – Trondheim
Norwegian University of
Science and Technology

Wavelet Techniques in Medical Imaging

Classification of UltraSound Images using the
Windowed Scattering Transform

Henrik Nicolay Finsberg

Master of Science in Physics and Mathematics
Submission date: June 2014
Supervisor: Yurii Lyubarskii, MATH
Co-supervisor: Ole Christensen, Technical University of Denmark
Preben Gråberg Nes, Sogn og Fjordane University College

Norwegian University of Science and Technology
Department of Mathematical Sciences

Abstract

In this thesis we will study wavelet techniques for image classification in ultrasound(US) images. The aim is to develop a method for classifying the degree of inflammation in finger-joints.

We develop and apply the techniques of the windowed scattering transform. This is a wavelet-based technique which is proven to be very efficient in image classification problems. Both theoretical and numerical sides have been considered. We also discuss other possible techniques for classification of US images, in particular a method based on the area of inflammation.

Preface

This thesis is written as a part of the Nordic Five Tech Master's Programme in Applied and Engineering Mathematics in the spring 2014. This thesis marks the end of a five year study, with three years at the masters program in Physics and Mathematics at NTNU, and two years at the Nordic masters program in Applied and Engineering Mathematics. The main supervisor is Professor Yurii Lyubarskii from the Department of Mathematical Sciences at NTNU. The co-supervisors are Professor Ole Christensen from the Department of Applied Mathematics and Computer Science at the Technical University of Denmark, and Associate Professor Preben Gråberg Nes from Sogn og Fjordane University College.

During the writing of this thesis I experienced the greatest moment of my life so far. Two months into the writing, my girlfriend gave birth to a healthy daughter. This caused my work to halt for a one month period, and my working area to move from Trondheim to Oslo.

I would first of all give a huge thank to Elisabeth, who has spent many lonely days babysitting our daughter. I admire her for the patience and the support she has given me.

I would also very much like to thank Professor Yurii Lyubarskii for his excellent guidance and supervision during the work of this thesis.

A special thanks goes to Associate Professor Preben Gråberg Nes, who has given me helpful feedback on drafts of my thesis, and provided me with images.

Finally, I would like to thank the Department of Mathematical Sciences at NTNU for providing me with the tools to study the beautiful theory of mathematical sciences.

Henrik Nicolay Finsberg
Trondheim, June 2014

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In this thesis we will study methods for classification of ultrasound(US) images of finger-joints. Medical doctors use these images to see if a patient can be diagnosed with synovitis. Synovitis is a type of inflammation occurring in finger-joints. The diagnose is divided into classes depending on the severity of the inflammation. US images are provided from a research project called Medusa, whose purpose is to develop a software for recognition and classification of synovitis.

To study these images we suggest the approach related to the windowed scattering transform. This modern technique was first published in 2012 by Stéphane Mallat and his research group at Ecole Polytechinqe, and has proven to be very efficient in a broad range of classification problems. Another method for classification based on the area of inflammation will also be discussed.

The thesis is structured in the following way: The first part contains a review of the theory. A background for the project is given in Chapter 2. Here we present the medical aspects of this project, and problems with existing classification methods. We give a general introduction to the classification methods studied in this thesis, and compare the scattering method with a human vision approach. In Chapter 3 we give a presentation of wavelet theory in one and two dimensions. This chapter introduces scattering wavelets, which are the building blocks for the windowed scattering transform. Application to edge-detection will also be discussed. Edge-detectors will be used as a tool to estimate the area of inflammation in the images. The windowed scattering transform will be introduced in Chapter 4. Outline of the construction and a pure analytic analysis will be provided. We will prove that the windowed scattering transform provides a representation which is locally translation invariant, stable to additive noise and stable to deformations. In addition, a proof that the energy is concentrated along frequency-decreasing paths will be presented.

In the second part we give a formulation of the theory applied to digital images. This is covered in Chapter 5. We will study the discrete version of the windowed scattering transform. In the end of this chapter we will show how statistical methods such as principal component analysis(PCA) may be applied in order to classify images.

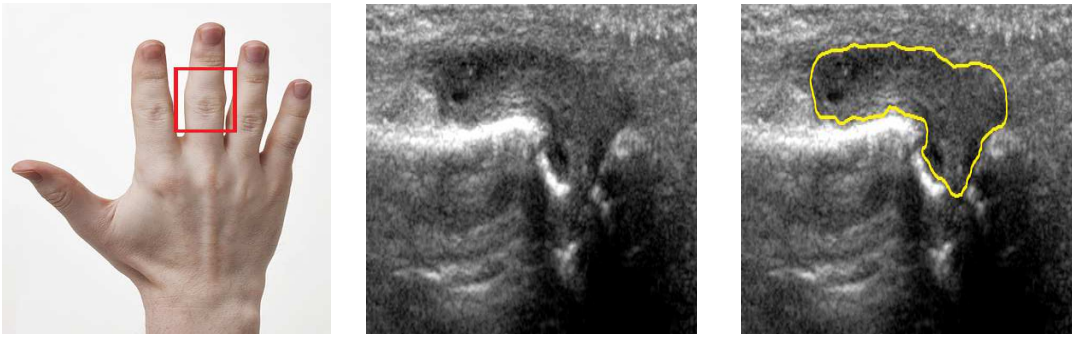
In the final part we present the results from the classification algorithms. This includes an analysis of optimal parameters and an evaluation of the methods. The

results show that scattering coefficients provides an efficient way to classify US images based on the degree of inflammation.

Chapter 2

Background

2.1 Synovitis



(a) Illustration of region where US images are taken.

(b) US image of finger joint.

(c) US image of finger joint. Contour of inflammation is highlighted.

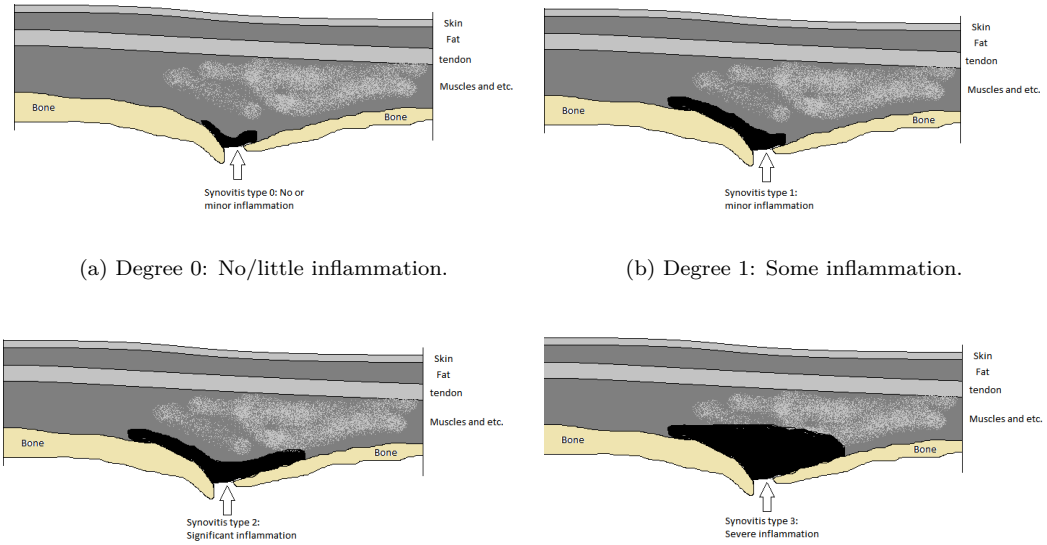
FIGURE 2.1: Illustration of region where US images are taken, and a typical US image of synovitis.

Joint disorders involving inflammation is called arthritis. Examples of such disorders are osteoarthritis and rheumatoid arthritis. When the inflammation occurs in the synovial membrane, which is the soft tissue found between the most movable joints, the inflammation is called synovitis. Synovitis may cause pain, limit the movement of the joints, and eventually erosion of the joint surface may cause loss of functioning. According to the US National Library of Medicine, as much as 6% of the population in the UK suffers from synovitis. The prevalence is highest among the elderly.

To detect synovits, US images are taken in the region illustrated in Figure 2.1(a). Figure 2.1(b) shows a typical US image of synovitis, and the contour of the inflammation is highlighted in Figure 2.1(c).

The inflammation is classified into four categories depending on the degree of inflammation:

0. No/little inflammation



(a) Degree 0: No/little inflammation.

(b) Degree 1: Some inflammation.

(c) Degree 2: Inflammation.

(d) Degree 3: Severe inflammation.

FIGURE 2.2: Phantom images of typical development of synovitis from degree 0 to degree 3.

1. Some inflammation
2. Inflammation
3. Severe inflammation

Figure 2.2 illustrates phantom images of the typical development of synovitis from degree 0 to 3. The black area represents the inflammation. Figure 2.2(a) shows an image with inflammation degree 0. Here one often sees a tiny vascular region in the finger-joint. For synovitis of degree 1 (Figure 2.2(b)), the inflammation expands along the right finger bone, whereas for degree 2 (Figure 2.2(c)) the inflammation expands along the finger bone in both directions. For degree 3 (Figure 2.2(d)) there is also expansion towards the fat and tendon.

The shape and area of the inflamed region characterize a particular degree of inflammation. For example, an image with degree 1 should have the same shape and area as the phantom image in Figure 2.2(b). However, variations should be expected.

2.1.1 Medical challenges

Rheumatologists, which are medical doctors specialized rheumatic diseases, diagnose a patient with synovitis based on an analysis of the US images. These rheumatologists are well trained to recognize synovitis. However, they are encountering the following problems when trying to detect and classify the degree of inflammation:

- A diagnose given by a medical doctor may not be validated/confirmed.

- A diagnose given by a medical doctor is highly subjective, meaning that different doctors may give different diagnoses based on the same image.

It is therefore preferable to develop an independent procedure for validating a diagnosis, so that medical doctors can minimize medication and treatment errors.

2.1.2 Available images

US images are provided from a research project called Medusa. The goal of this project is to develop a software which can be used by medical doctors as a tool to determine the right diagnosis involving synovitis. The total amount of available images from this project is well over 2000. However, this project is still in an early phase, so that only a few images have been analyzed. To test our classification methods, we need information about the inflammation degree in all the images we are testing, so that we can say whether images are classified correctly by our algorithm. The total amount of classified images are 296. Moreover, for each image there is a corresponding image where the inflamed region is annotated, such as in Figure 2.1(c).

2.2 Recognition based on human vision

To classify images, we need to be able to recognize features in the images which are specific within each class. A conjecture proposed in [PMA+12] states the following about object recognition:

> *The "main" difficulty of recognition, in the sense of sample complexity, of object categorization is due to all the transformations that the image is usually subject to: translation, scale, illumination and rotations.*

This conjecture implies that if all images of objects are rectified with respect to these transformations, then object recognition and classification are easy. In [PMA+12] they argue that the ventral stream in the human brain is invariant to these transformations. Hence, by studying how the ventral stream processes images, we may get a hint on how to develop an efficient algorithm for image classification.

2.2.1 Visual perception

In this section we will give a brief introduction to how the human brain processes images, and how it recognizes objects. An illustration of this process is provided in Figure 2.3. The theory is taken from [KSJS00].

Light emitted by objects is sent through the lenses of the eye, and onto the retina at the back end of the eye. The retina consists of many receptive fields, which again consists of many receptors. Each receptive field is connected to one neuron called a ganglion cell, who's job is to transmit information from the eye to the brain. When light hits the retina, it activates these receptors. Each combination of receptors in the receptive field causes the ganglion cell to fire a unique nerve impulse which travels along fibres called LGN, all the way to the visual cortex.

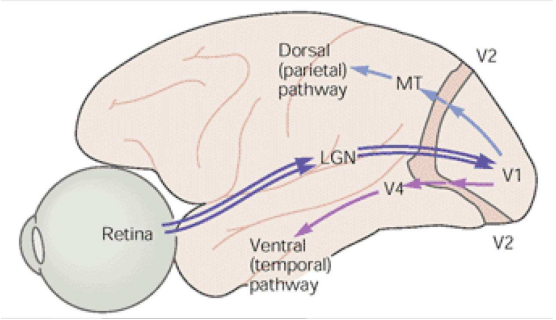


FIGURE 2.3: Illustration of how the brain process images: Ganglion cells transmit signal from the retina to the primary visual cortex V1, through LGN fibres. The signal is processed and sent along two pathways, the dorsal pathway and the ventral pathway.

The visual cortex, which is located in the back of the brain, is the part responsible for processing visual information. The visual cortex is divided into the primary visual cortex V1, and the extra-striate visual cortical areas, V2 -V5. The information sent from the retina is mapped to a grid in the primary visual cortex consisting of hypercolumns. Each hypercolumn analyzes information from a small region of the retina, and extracts information about stereopsis(visual depth), color and orientation of line segments.

The analysis of orientation of line segments is done by simple and complex cells. Simple cells are highly specialized and will be activated by a particular orientation in the image. In 1985, John Daugman discovered that simple cells, could be modelled as two-dimensional Gabor wavelets [Dau85]. Several simple cells with the same orientation converge together and form complex cells.

After the signal has been analyzed by these hypercolumns, it is sent further down two ”pathways”. One of these pathways, known as the ventral pathway, or ventral stream, is the process associated with object recognition and classification. This path starts in V1, goes through V2 and V4 and ends in the temporal lobe. This is where for instance the hippocampus is located, and is associated with visual memories.

The key to understand how the brain is able to recognize specific objects, lies in the ventral stream. What really happens is still an open question. What they do know is that some kind of invariant structure is being built. As an example, the human brain recognizes a specific face, despite changes in viewpoint, illumination or expression. Change of viewpoint may be formulated as a translation of the signal, a change in expression is the same as a small deformation of the signal, and a change in the illumination can be viewed as additive noise. Formally, we can say that the brain is invariant, or at least stable under these transformations. This is discussed in more detail in [PMA+12].

Below we formulate the main steps of the image recognition process performed by the human brain. We also propose a corresponding image recognition algorithm which could be implemented on a computer.

Brain	Computer
1. Image is captured by the eye and sent to the retina. 2. Receptors are activated and transmit the signal to the visual cortex. The signal is mapped onto a grid formed by hypercolmns. 3. Simple cells extract information about orientation of line segments. 4. Information is sent along the ventral stream, and invariant structure is built.	1. Consider an image function f. 2. Digitalize image f so that it is accessible for a computer. 3. Using two-dimensional Gabor wavelets, calculate the wavelet transform of the image f. 4. Apply the windowed scattering transform.

As we will see, this procedure has many similarities with the approach studied in this thesis. What happens in the ventral stream is not fully understood, however the windowed scattering transform may provide a partial answer.

2.3 Classification methods

We treat grey images as continuous functions in $L^2(\mathbb{R}^2)$. If $f \in L^2(\mathbb{R}^2)$ is an image, then the value $f(x, y)$ represents the intensity at the spatial coordinate (x, y). To classify images we need some kind of metric to measure the distance between them. It should be so that images which belong to the same class are close, and images which belong to different classes are distant. The idea is to look at the variability within each class, and try to find a representation of these signals where this variability is reduced. In order to do so, we construct a Hilbert space $\mathcal{H}$, and an operator $\Phi : L^2(\mathbb{R}^2) \to \mathcal{H}$. The operator should be so that if f and g belong to the same class then the distance $\|\Phi(f) - \Phi(g)\|_{\mathcal{H}}$ is small, where as if they belong to different classes, the distance should be large.

2.3.1 Classification based on the area of inflammation

The first method builds on the fact that the area of the inflamed region is dependent on the degree of inflammation, i.e that larger area means higher degree of inflammation. In this case, one should be able to classify images based on the area

of the inflammation, that is

$$\Phi(f) = \text{area of inflammed region.}$$

As the area is just a positive real number, the Hilbert space $\mathcal{H}$ will be the Euclidean space $\mathbb{R}$, with the usual inner product.

An edge-detector is applied to locate the boundary of the inflamed region. Some problems related to detection of the correct boundary, and how these problems are solved will be discussed in Section 5.2.1. With information about the boundary, the area may be calculated by using a discrete version of Greens formula. Predication intervals are computed for each class, and a new image is classified according to minimum distances to these intervals.

An edge-detector will be presented in Section 3.2. This edge-detector uses wavelets to detect sharp transitions in signals. An introduction to wavelets is given in section 3.1.

2.3.2 Classification based on scattering coefficients

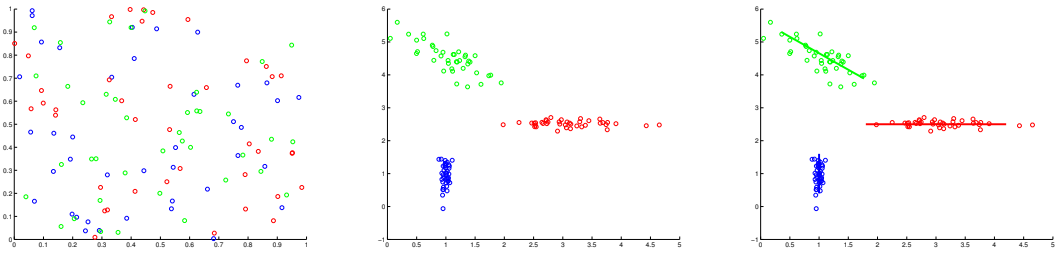
The other method builds on the same principles as how the brain recognizes objects. Based on visual perception we know that the brain perceives two objects as the same if one is translated, slightly deformed or illuminated.

This variability can be reduced by considering the scattering coefficients. Scattering coefficients provide a representation of an image which is locally translation invariant, stable to additive noise and stable to the action of diffeomorphisms. They are computed by cascading wavelet transforms with a non-linear modulus along different scales and orientations. The output is then averaged by a low-pass filter whose support is proportional to the amount of translation invariance. In Chapter 4 we will present the windowed scattering transform which transforms an image into its scattering coefficients. The building blocks of this transformation are scattering wavelets, which will be presented in Section 3.3. The idea behind the windowed scattering transform is due to Stéphane Mallat and his research group at Ecole Polytechinqe. The main theory is taken from the article [Mal12] and the doctoral thesis [Bru12].

The output from the windowed scattering transform is a family of functions in $L^2(\mathbb{R}^2)$, so that the Hilbert space $\mathcal{H}$ will be the product space generated by copies of $L^2(\mathbb{R}^2)$. When applying the windowed scattering transform to digital images, the output will be a finite number of discrete functions. In this case the Hilbert space $\mathcal{H}$ becomes $\mathbb{R}^N$ for some $N \in \mathbb{N}$. A survey of how to compute the scattering coefficients of digital images is presented in Chapter 5.

If each image is represented as a point in the Hilbert space $\mathcal{H}$, each class can be represented as a regular manifold. To classify images, we will approximate each regular manifold by an affine space based on principle component analysis(PCA). This statistical method uses eigenvectors of the variance-covariance matrix to approximate the scattering coefficients. For each class there will thus be a corresponding affine space.

In Figure 2.4, an illustration of this process with three classes is provided. Each image is represented by a point in the plane, and images belonging to the same



(a) Singles before any processing.

(b) Signals after applying Φ.

(c) Approximation of subspaces through e.g. PCA.

FIGURE 2.4: Illustration of the steps in image classification.

class have the same color. In Figure 2.4(a), the images are seemingly randomly distributed. When applying the windowed scattering transform, images belonging to the same class are clustered together, see Figure 2.4(b). In Figure 2.4(c) we see the approximation of these clusterings by affine spaces(in this case, lines).

To classify a new image, one classifies it based on projections on these affine spaces. The class corresponding to the affine space that is closest to the new image will be the class for which we classify this new image.

Mallat and his research group have developed a toolbox for Matlab called ScatNet [ASM+14] which we will use to test this method on US images of finger-joints.

In the next chapter we will present our main tool in the quest of constructing the operator Φ, namely wavelets.

Chapter 3

Wavelets

This chapter surveys some basic wavelet theory. The first chapter introduces the wavelet transform in one and two dimensions.

Applications to edge-detection will be covered in Section 3.2, where we present the Canny edge-detection algorithm [Can86]. The last section introduces scattering wavelets. These wavelets will be the building blocks for the windowed scattering transform which will be defined in Chapter 4.

The theory for the one-dimensional wavelet transform is mainly taken from [Mal09]. When constructing the scattering wavelets in Section 3.3, one wavelet is dilated and translated along different orientations. As this construction is also applicable to the wavelet used in the Canny edge-detection algorithm, we will limit ourself to this construction. The main theory for two-dimensional wavelets is taken from [AMVA04].

3.1 Introduction

We will start with wavelet theory for one-dimensional signals, and thereafter extend this theory to two dimensions.

3.1.1 Wavelets in one dimension

Definition 3.1. *[Mal09, p.102] A one-dimensional wavelet is a function* $\psi \in L^1(\mathbb{R}) \cap L^2(\mathbb{R})$ *with zero average*

$$\int_{\mathbb{R}} \psi(x)dx = 0. \tag{3.1}$$

A dilated and translated wavelet is written

$$\psi_{u,s}(x) = s^{-1/2}\psi\left(\frac{x-u}{s}\right), \quad (u,s) \in \mathbb{R} \times \mathbb{R}^+. \tag{3.2}$$

The factor $s^{-1/2}$ ensures that $\|\psi_{u,s}\|_{L^2} = \|\psi\|_{L^2}$. The one-dimensional wavelet transform is obtained as the inner product of a signal f with a dilated and translated wavelet ψ.

Definition 3.2. *[Mal09, p.102] The wavelet transform of $f \in L^2(\mathbb{R})$ at position u and scale s is*

$$Wf(u,s) = \langle f, \psi_{u,s} \rangle = \int_{\mathbb{R}} f(x) s^{-1/2} \psi^* \left(\frac{x-u}{s} \right) dx, \tag{3.3}$$

where $\psi^*(x)$ denotes the complex conjugated of $\psi(x)$.

If we define $\bar{\psi}_s(x) = s^{-1/2}\psi^*\left(\frac{-x}{s}\right)$, then we see that the wavelet transform can be written as a convolution,

$$Wf(u,s) = \int_{\mathbb{R}} f(x) s^{-1/2} \psi^* \left(\frac{x-u}{s} \right) dx = \int_{\mathbb{R}} f(x) \bar{\psi}_s(u-x) dx = f * \bar{\psi}_s(u). \tag{3.4}$$

The wavelet transform is invertible if the wavelet satisfies the admissibility condition given in Definition 3.3.

Definition 3.3. *[AMVA04, p.6] A one-dimensional wavelet ψ is called admissible if*

$$C_\psi = \int_{-\infty}^{\infty} \frac{|\widehat{\psi}(\omega)|^2}{|\omega|} d\omega < +\infty. \tag{3.5}$$

Here $\widehat{\psi}$ denotes the Fourier transform of the wavelet ψ. Most authors define a wavelet as a function which satisfies the admissibility condition. We will however follow the definition given in [Mal09].

Example 3.1. An important wavelet in edge-detection is the first derivative of the Gaussian,

$$\psi(x) = \frac{x}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}. \tag{3.6}$$

Since

$$\widehat{\psi}(\omega) = -i\omega e^{-\frac{\omega^2}{2}},$$

we see that

$$\begin{aligned} C_\psi &= \int_{-\infty}^{\infty} \frac{|\widehat{\psi}(\omega)|^2}{|\omega|} d\omega = \int_{-\infty}^{\infty} |\omega| e^{-\omega^2} d\omega \\ &= 2\int_0^{\infty} \omega e^{-\omega^2} d\omega = \int_0^{\infty} e^{-\xi} d\xi = 1 < \infty. \end{aligned}$$

Hence ψ is an admissible wavelet.

3.1.2 Wavelets in two dimensions

As in one dimension, a two-dimensional wavelet will be defined as a function in $L^2(\mathbb{R}^2) \cap L^1(\mathbb{R}^2)$ with zero average.

Definition 3.4. *[AMVA04, p.34] A two-dimensional wavelet is a function $\psi \in L^1(\mathbb{R}^2) \cap L^2(\mathbb{R}^2)$ with zero average*

$$\int_{\mathbb{R}^2} \psi(x) dx = 0. \tag{3.7}$$

The wavelet transform will be constructed by dilating, rotating and translating the wavelet. A dilated, rotated and translated wavelet is written

$$\psi_{u,s,\theta}(x) = s^{-1}\psi\left(r_\theta^{-1}\left(\frac{x-u}{s}\right)\right), \quad (s,u,r_\theta) \in \mathbb{R}_+ \times \mathbb{R}^2 \times SO(2), \tag{3.8}$$

where

$$SO(2) = \left\{ r_\theta = \begin{pmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{pmatrix} : \theta \in [0, 2\pi) \right\}.$$

The normalization is chosen with respect to the L^2-norm, i.e $\|\psi_{u,s,\theta}\|_{L^2} = \|\psi\|_{L^2}$.

Definition 3.5. *[AMVA04, p.36] The wavelet transform of $f \in L^2(\mathbb{R}^2)$ at position u, scale s, and orientation θ is*

$$Wf(u,s,\theta) = \langle f, \psi_{u,s,\theta}\rangle = \int_{\mathbb{R}^2} f(x) s^{-1}\psi^*\left(r_\theta^{-1}\left(\frac{x-u}{s}\right)\right) dx. \tag{3.9}$$

The two dimensional admissibility condition is given in Definition 3.6.

Definition 3.6. *[AMVA04, p.34]A two-dimensional wavelet ψ is called admissible if*

$$C_\psi = \int_{\mathbb{R}^2} \frac{|\widehat{\psi}(\omega)|^2}{|\omega|^2} d\omega < +\infty. \tag{3.10}$$

Translation, dilation and rotation are unitary operations, and will therefore preserve the admissibility condition. Hence, if ψ is a wavelet then $\psi_{u,s,\theta}$ is again a wavelet. Moreover, the linear span of the family $\{\psi_{u,s,\theta} : (s,u,r_\theta) \in \mathbb{R}_+ \times \mathbb{R}^2 \times SO(2)\}$ is a dense subspace of $L^2(\mathbb{R}^2)$ [AMVA04, p.35].

The admissibility condition (3.10) ensures that one can reconstruct the original signal from its wavelet transform. For admissible wavelets we have the reconstruction formula

$$f(x) = \frac{1}{C_\psi} \int_{\mathbb{R}_+ \times \mathbb{R}^2 \times SO(2)} \psi_{u,s,\theta}(x) Wf(u,s,\theta) du \frac{ds}{s^3} d\theta. \tag{3.11}$$

If one inserts this formula into the wavelet transform (3.9) one gets the reproduction property

$$Wf(u_0,s_0,\theta_0) = \int_{\mathbb{R}_+ \times \mathbb{R}^2 \times SO(2)} K(u,u_0,s,s_0,\theta,\theta_0) Wf(u,s,\theta) du \frac{ds}{s^3} d\theta. \tag{3.12}$$

The kernel $K(u,u_0,s,s_0,\theta,\theta_0) = C_\psi^{-1}\langle \psi_{u,s,\theta}, \psi_{u_0,s_0,\theta_0}\rangle$ is called the reproducing kernel. This property implies that the wavelet transform is a highly redundant representation. Indeed, with knowledge about the wavelet transform at a given point (u,s,θ) one can find the wavelet transform in neighbouring points (u_0,s_0,θ_0), with $K(u,u_0,s,s_0,\theta,\theta_0) \neq 0$. Therefore one can restrict the wavelet transform to a subset of discrete points without any loss of information.

First, consider a finite group of rotations, $G \subset SO(2)$, e.g for fixed $K \in \mathbb{N}$,

$$G = \left\{ r_k : r_k = \begin{pmatrix} \cos\theta_k & -\sin\theta_k \\ \sin\theta_k & \cos\theta_k \end{pmatrix}, \theta_k = \frac{2\pi k}{K}, k = 0, 1 \cdots K-1 \right\}, \tag{3.13}$$

and write

$$\psi_{u,s,\theta_k}(x) = \psi^k_{u,s}(x), \quad \text{and} \quad W^k f(u,s,\theta) = \langle f, \psi^k_{u,s} \rangle.$$

Proposition 3.7. *[AMVA04, p.66] Let ψ be a two-dimensional wavelet, and $K \in \mathbb{N}$. If there exist two constants $0 < A \leq B < \infty$ such that*

$$\forall \omega \in \mathbb{R}^2 - \{(0,0)\}, \quad A \leq \sum_{k=1}^{K} \sum_{j=-\infty}^{\infty} |\widehat{\psi^k}(2^j\omega)|^2 \leq B, \tag{3.14}$$

then

$$A\|f\|^2_{L^2} \leq \sum_{j=-\infty}^{\infty} \sum_{k=1}^{K} 2^{-2j} \|W^k f(\cdot, 2^j)\|^2_{L^2} \leq B\|f\|^2_{L^2}. \tag{3.15}$$

Moreover, define $\tilde{\psi}^k$ via the relation

$$\sum_{j=-\infty}^{\infty} \sum_{k=1}^{K} \widehat{\psi^k}^*(2^j\omega)\widehat{\tilde{\psi}^k}(2^j\omega) = 1, \quad \forall \omega \in \mathbb{R}^2 - \{(0,0)\}. \tag{3.16}$$

Then the following reconstruction formula holds

$$f(x) = \sum_{j=-\infty}^{\infty} \sum_{k=1}^{K} 2^{-2j} (W^k f(\cdot, 2^j) * \tilde{\psi}^k_{2^j})(x). \tag{3.17}$$

Proof. First note that if $\bar{\psi}^k_s(x) = \psi^{k*}_s(-x)$, then

$$W^k f(u,s) = \langle f, \psi^k_{u,s} \rangle = f * \bar{\psi}^k_s(u).$$

Let

$$\phi^k_j(\omega) = \widehat{W^k f(\cdot, 2^j)}(\omega) = \widehat{\bar{\psi}^k_{2^j}}(\omega)\widehat{f}(\omega).$$

The Fourier transform $\widehat{\bar{\psi}^k_{2^j}}(\omega)$, will be computed in detail,

$$\widehat{\bar{\psi}^k_{2^j}}(\omega) = \frac{2^{-j}}{2\pi} \int_{\mathbb{R}^2} \psi^*(-2^{-j} r^{-1}_{\theta_k} x) e^{-i\langle \omega, x \rangle} dx.$$

With the change of variables, $y = -2^{-j} r^{-1}_{\theta_k} x$, the Jacobian matrix is given by

$$D = \begin{pmatrix} -2^{-j}\cos\theta_k & 2^{-j}\sin\theta_k \\ -2^{-j}\sin\theta_k & -2^{-j}\cos\theta_k \end{pmatrix}, \quad \det D = 2^{-2j}.$$

Hence

$$\widehat{\bar{\psi}^k_{2^j}}(\omega) = \frac{2^j}{2\pi}\int_{\mathbb{R}^2} \psi^*(y)e^{i\langle\omega,2^j r_{\theta_k} y\rangle}dy = \left(\frac{2^j}{2\pi}\int_{\mathbb{R}^2}\psi(y)e^{-i\langle\omega,2^j r_{\theta_k} y\rangle}dy\right)^*.$$

The adjoint of a rotation matrix r_{θ_k} equals its inverse, hence

$$\widehat{\bar{\psi}^k_{2^j}}(\omega) = 2^j\left(\frac{1}{2\pi}\int_{\mathbb{R}^2}\psi(y)e^{-i\langle 2^j r^{-1}_{\theta_k}\omega,y\rangle}dy\right)^* = 2^j\widehat{\psi^k}^*(2^j\omega). \tag{3.18}$$

Consequently

$$\phi^k_j(\omega) = 2^j\widehat{\psi^k}^*(2^j\omega)\widehat{f}(\omega).$$

Applying condition (3.14) shows that

$$A|\widehat{f}(\omega)|^2 \le \sum_{j=-\infty}^{\infty}\sum_{k=1}^{K} 2^{-2j}|\phi^k_j(\omega)|^2 \le B|\widehat{f}(\omega)|^2.$$

Integration with respect to ω and applying Parevals equality(A.11) yields (3.15). Taking the Fourier transform on both sides of (3.17) gives

$$\widehat{f}(\omega) = \sum_{j=-\infty}^{\infty}\sum_{k=1}^{K}\widehat{f}(\omega)\widehat{\psi^k}^*(2^j\omega)\widehat{\tilde{\psi}^k}(2^j\omega).$$

Condition (3.16) implies that (3.17) indeed holds. □

Condition (3.14) implies that restricting the wavelet transform to dyadic scales gives possibility to reconstruct the original signal from its wavelet transform. Moreover, the upper bound $B < \infty$, insures that the map $f \mapsto \{2^{-j}W^k f(\cdot, 2^j)\}$ is continuous. The lower bound $A > 0$, implies that the coefficients $\{2^{-j}W^k f(\cdot, 2^j)\}$ are numerically stable.

One way to interpret condition (3.14) is that the two-dimensional Fourier plane needs to be covered by dyadic dilations of $\widehat{\psi^k}$ for $k = 1, 2, \cdots, K$. A two-dimensional wavelet transform, with a wavelet satisfying condition (3.14) will be referred to as a stable representation.

The following example will be important in the next section where we will see how wavelets can be used to detect edges in an image.

Example 3.2. For $x = (x_1, x_2) \in \mathbb{R}^2$, let

$$\psi(x) = \frac{\langle x, (1,0)\rangle}{2\pi}e^{-\frac{|x|^2}{2}}. \tag{3.19}$$

As ψ is antisymmetric, it is easy to verify that it has zero average, and hence is a wavelet. The wavelet will be rotated by 0 and $\frac{\pi}{2}$ radians. Let

$$r_1 = \begin{pmatrix} \cos(0) & -\sin(0) \\ \sin(0) & \cos(0) \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \qquad r_1^{-1} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix},$$
$$r_2 = \begin{pmatrix} \cos(\frac{\pi}{2}) & -\sin(\frac{\pi}{2}) \\ \sin(\frac{\pi}{2}) & \cos(\frac{\pi}{2}) \end{pmatrix} = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}, \quad r_2^{-1} = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix},$$

and define

$$\psi^1(x) = \psi(r_1^{-1}x) = \frac{x_1}{2\pi} e^{-\frac{|x|^2}{2}} \tag{3.20}$$
$$\psi^2(x) = \psi(r_2^{-1}x) = \frac{x_2}{2\pi} e^{-\frac{|x|^2}{2}}. \tag{3.21}$$

Since $\widehat{\psi^k}(\omega) = -i\omega_k e^{-\frac{|\omega|^2}{2}}$, the series

$$\sum_{j=-\infty}^{\infty} |\widehat{\psi^k}(2^j\omega)|^2 = \sum_{j=-\infty}^{\infty} |2^j\omega_k|^2 e^{-2^{2j-1}|\omega|^2}, \quad k = 1, 2$$

is convergent for all $\omega \in \mathbb{R}^2$. Moreover, since $|2^j\omega|^2 e^{-2^{2j-1}|\omega|^2} > 0$ for all $\omega \in \mathbb{R}^2 - \{(0,0)\}$, the series is bounded from below by a positive constant A for any $\omega \in \mathbb{R}^2 - \{(0,0)\}$. Hence we have a stable representation.

3.2 Wavelets in edge detection

The main goal of this thesis is to develop a method for classifying US images of finger-joints according to the degree of synovits. The degree of this inflammation depends on the shape and area of the inflamed region. It is therefore interesting to see if it is possible to classify images based on the area. To find this area we will use an edge-detection algorithm to detect the boundary of the inflamed region. The area can the be calculated by using Greens formula.

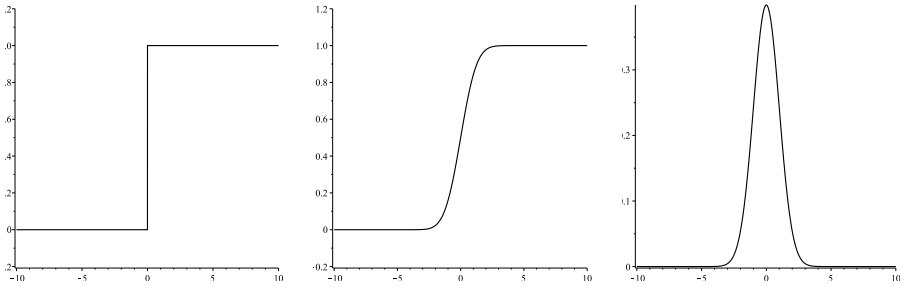
We will first explain how wavelets are able to detect edges in one-dimensional signals.

3.2.1 Catching edges

Edge-points are points where the function changes rapidly. We know that rate of change is characterized by the derivative of the function. In other words, we will look for local maxima and minima of the derivative. Consider the Heaviside function

$$H(x) = \begin{cases} 1 & \text{if } x \geq 0, \\ 0 & \text{if } x < 0, \end{cases} \tag{3.22}$$

which is plotted in Figure 3.1(a). This function has a sharp transition at the origin, but as it is discontinuous it is not differentiable everywhere. To overcome



(a) $H(x)$. (b) $(H * \theta)(x)$. (c) $\frac{d}{dx}(H * \theta)(x)$.

FIGURE 3.1: Plot of the Heaviside function, a smoothed heaviside function and the derivative of the smoothed function.

this problem, we can exploit a useful property of the convolution operation, namely if $f \in L^2(\mathbb{R})$ and $g \in C^n(\mathbb{R})$, then their convolution is n times differentiable (A.9).

Let θ be the Gaussian convolution kernel,

$$\theta(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}. \tag{3.23}$$

Since $\theta \in C^\infty(\mathbb{R})$, the convolution $H * \theta(x)$ is differentiable.

In Figure 3.1(b) we see a plot of the function $H * \theta(x)$. The derivative $\frac{d}{dx}(H * \theta)(x)$ is illustrated in Figure 3.1(c). From this we see that the derivative has a maximum at the origin which indicates that we have an edge located there.

Another property of the convolution operation is that (A.9),

$$\frac{d}{dx}(f * \theta)(x) = f * \theta'(x).$$

We identify the wavelet

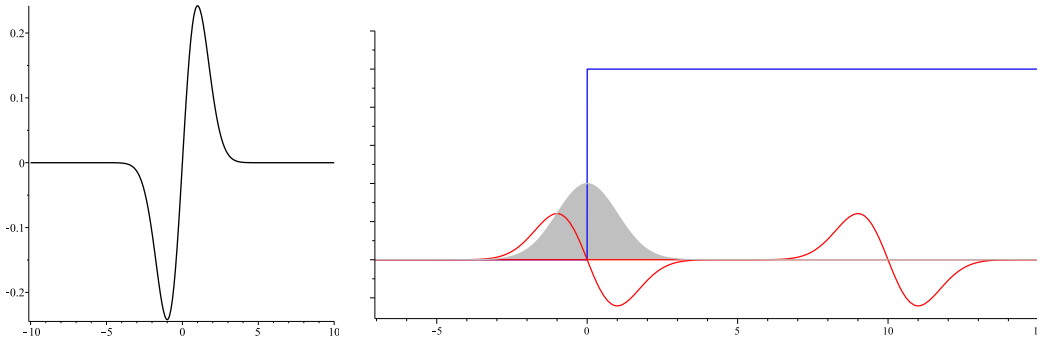
$$\psi(x) = -\frac{d}{dx}\theta(x) = \frac{x}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}. \tag{3.24}$$

This is the same wavelet as the one defined in Example 3.1. The corresponding wavelet transform may be written as

$$Wf(u, s) = f * \bar{\psi}_s(u) = s\frac{d}{du}(f * \bar{\theta}_s)(u),$$

where $\bar{\theta}_s(x) = s^{-1/2}\theta\left(\frac{-x}{s}\right) = s^{-1/2}\theta\left(\frac{x}{s}\right)$.

In Figure 3.2(a) we see the graph of the wavelet ψ. In Figure 3.2(b) we see the Heaviside function and the wavelet at scale $s = 1$, translated at position $x = 0$ and $x = 10$. At a given point $u \in \mathbb{R}$, the convolution can be viewed as the area under the graph $f(x)\psi(x-u)$. As $\psi(x)$ is concentrated around $x = 0$, the product $f(x)\psi(x-u)$ is non-negligible only at an interval around $x = u$. If f is constant at that interval, the integral will be approximately zero since the wavelet has zero



(a) $\psi(x) = -\frac{d}{dx}\theta(x)$.

(b) Visualization of how the wavelet can be used to detect edges in a signal.

FIGURE 3.2: Figure 3.2(a) shows a plot of the wavelet corresponding to the first derivative of the Gaussian. Figure 3.2(b) illustrates how this wavelet can be used for edge-detection.

average. This is the case when $u = 10$ in Figure 3.2(b). At $u = 0$, f has a rapid change. As ψ is odd, the difference between the function $f(x)\psi(x-u)$ at $u < 0$ and $u > 0$ will be large, and maximal at $u = 0$. The output of the wavelet transform is thus concentrated near the edge point.

Since the wavelet transform may be negative or complex, edges are detected by searching for local maxima of the wavelet modulus $|Wf(u, s)|$.

Definition 3.8. *[Mal09, p.231] A wavelet modulus maximum is defined as a point (u_0, s_0), where $|Wf(u, s_0)|$ has a local maximum at $u = u_0$.*

In [Can86], edge points are thus defined in the following way:

Definition 3.9. *[Can86] Let $f \in L^2(\mathbb{R})$ and let θ be the Gaussian convolution kernel defined in 3.23. A point x_0 is said to be an edge point at scale s, if the derivative $|\frac{d}{dx}(f * \theta_s)(x)|$ has local maximum at x_0.*

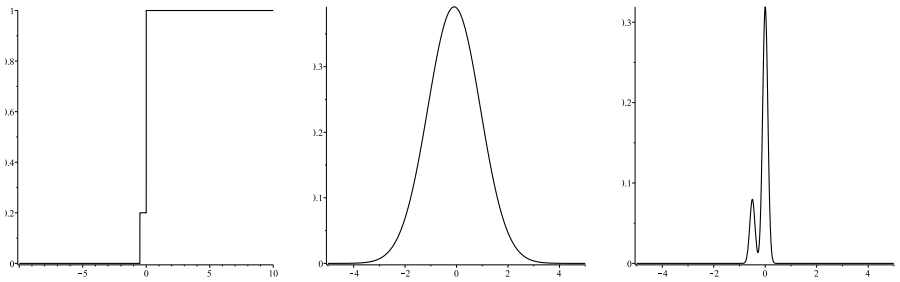
This implies that the edge-points are the wavelet modulus maxima computed with the wavelet defined in (3.24).

An advantage of introducing the wavelet concept is that we can vary both the spatial variable u, and the scale variable s. By varying the scale variable we can zoom in on the function, and detect finer structure. Let us illustrate this with an example.

Example 3.3. Consider the signal, illustrated in Figure 3.3(a),

$$f(x) = \begin{cases} 1 & \text{if } x \geq 0, \\ 0.2 & \text{if } -0.5 \leq x < 0, \\ 0 & \text{if } x < -0.5. \end{cases} \tag{3.25}$$

This function has an edge point at $x = 0$ and $x = -0.5$. At a coarse scale, that is for large values of s, the wavelet transform detects the main features in the signal.



(a) $f(x)$.

(b) $Wf(u,s)$, for s large.

(c) $Wf(u,s)$, for s small.

FIGURE 3.3: A plot of the function (3.25), and the corresponding wavelet transform at coarse and fine scale.

Figure 3.3(b) shows that there is an edge close to the origin. However, by zooming in, i.e decreasing the scale, information about the additional edge is emerging, see Figure 3.3(c).

In noisy images such as ultrasound images, this zooming property is key to detect the correct edges.

3.2.2 Edge detection in images

The extension from one-dimensional signals to two-dimensional signals is straight forward. If $f : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ is a continuously differentiable function, let us denote the gradient of f as

$$\vec{\nabla} f = \left(\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2} \right),$$

and the directional derivative in the direction $\vec{n} = (n_1, n_2)$ is

$$\frac{\partial f}{\partial \vec{n}} = \vec{\nabla} f \cdot \vec{n} = \frac{\partial f}{\partial x_1} n_1 + \frac{\partial f}{\partial x_2} n_2, \quad |\vec{n}| = 1.$$

At any point (x_1, x_2), the direction of the maximal change is given by the direction of the gradient, and hence the directional derivative will be maximal if $\vec{\nabla} f$ and $\vec{n}$ are collinear.

To detect edges in a two-dimensional signal, one can use a similar approach as when detecting singularities in a one-dimensional signal. Let θ be the two-dimensional Gaussian function

$$\theta(x_1, x_2) = \frac{1}{2\pi} e^{-\frac{x_1^2 + x_2^2}{2}}. \tag{3.26}$$

Define two functions ψ^1 and ψ^2 as the negative of the partial derivatives of this function, see Example 3.2. If $\bar{\theta}_s = s^{-1}\theta(-s^{-1}x)$, then

$$\bar{\psi}_s^k(x) = s^{-1}\psi^k(-s^{-1}x) = \frac{\partial\theta}{\partial x_k}(-s^{-1}x) = \frac{\partial\bar{\theta}}{\partial x_k}(s^{-1}x) = s\frac{\partial\bar{\theta}_s}{\partial x_k}(x).$$

Hence

$$Wf(u,s) = \begin{pmatrix} W^1 f(u,s) \\ W^2 f(u,s) \end{pmatrix} = s\begin{pmatrix} \frac{\partial}{\partial u_1}(f * \bar{\theta}_s)(u) \\ \frac{\partial}{\partial u_2}(f * \bar{\theta}_s)(u) \end{pmatrix} = s\vec{\nabla}(f * \bar{\theta}_s)(u). \tag{3.27}$$

The modulus $Mf(u,s)$ will be the length of this vector

$$Mf(u,s) = \sqrt{|W^1 f(u,s)|^2 + |W^2 f(u,s)|^2}, \tag{3.28}$$

and the direction of the maximal change is given by the angle

$$Af(u,s) = \begin{cases} \alpha(u) & \text{if } W^1 f(u,s) \geq 0, \\ \pi + \alpha(u) & \text{if } W^1 f(u,s) < 0, \end{cases} \tag{3.29}$$

where

$$\alpha(u) = \tan^{-1}\left(\frac{W^2 f(u,s)}{W^1 f(u,s)}\right). \tag{3.30}$$

We define an edge point as a local maximum of Mf in the direction Af.

Definition 3.10. *[Can86] Let $f \in L^2(\mathbb{R}^2)$ and let θ be the two-dimensional Gaussian kernel defined in 3.26. A point $x_0 \in \mathbb{R}^2$ is said to be an edge point if it is a local maximum (in the direction $\vec{n}$) of the function $\vec{n} \cdot \vec{\nabla}(f * \bar{\theta}_s)(x)$. This implies that*

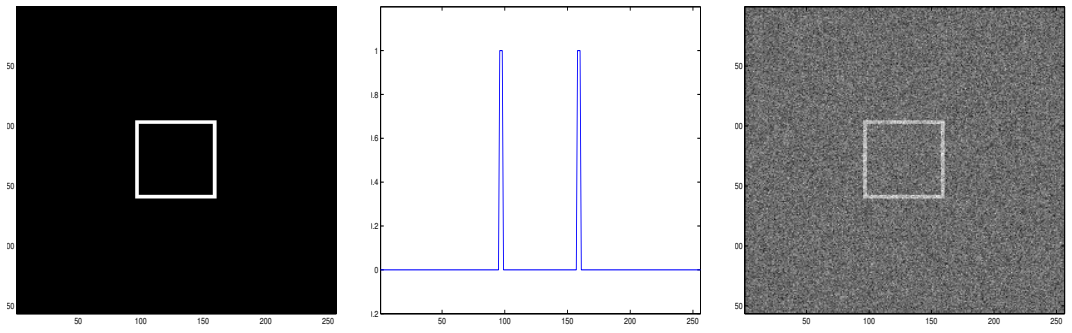
$$\frac{\partial}{\partial\vec{n}} Wf(u,s) = 0.$$

The direction $\vec{n}_s(u) = (\cos Af(u,s), \sin Af(u,s))$ will be the direction in which we have a modulus maxima. Thus the modulus maxima will be inflection points of $(f * \bar{\theta}_s)(u)$.

In 1986 John Canny[Can86] published an article where he presented this edge-detection algorithm. This algorithm will therefore be referred to as the Canny edge-detector.

The Canny edge-detector computes modulus maxima on a fixed scale s, and stores the position u of each modulus maxima together with $Mf(u,s)$ and $Af(u,s)$. Edge points with a low amplitude are usually caused by noise, or small transitions in the image. To detect significant edges a thresholding of the amplitude of the modulus maxima is applied.

When doing this kind of thresholding, variations in the amplitude along the boundary of an object may cause the boundary-line to break. This phenomena is called streaking. This is improved by introducing two different thresholds, T_{high} and T_{low}, with $T_{\text{low}} < T_{\text{high}}$. If the amplitude of the modulus maxima is above the high threshold T_{high}, then it should immediately be marked as an edge. If



(a) Noisefree Box.

(b) Cross section of box at the center.

(c) Box with Gaussian white noise.

FIGURE 3.4: Plot images from Example 3.4.

the amplitude of the modulus maxima is above the low threshold T_{low}, and in addition connected to a line segment in which some of the points are above the high threshold it should also be marked as an edge.

3.2.3 Detecting the boundary of the inflamed region

We want to apply the Canny edge-detector to detect the boundary of the inflammation in US images of finger-joints. US images usually contain a lot of noise due to the coherent nature of the ultrasound radiation. This noise can be viewed as a random process which is added to the image.

In the next example we will illustrate the use of the Canny edge-detector with a simple example, and also show that noise gives rise to some additional difficulties when trying to detect the correct edges.

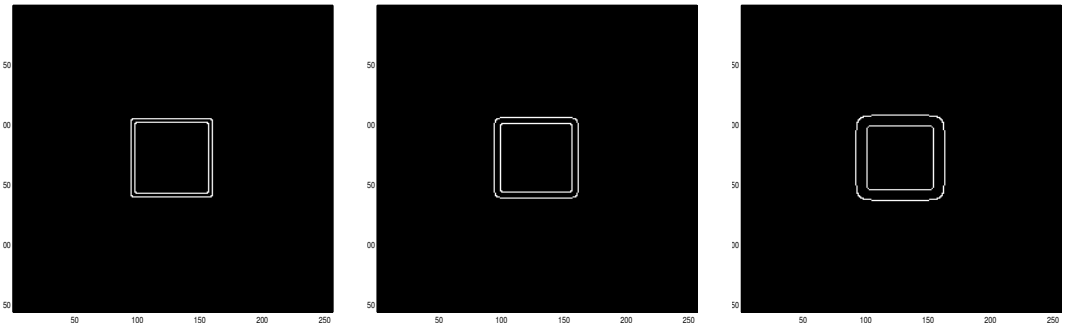
Example 3.4. Consider the box illustrated in Figure 3.4(a). The cross-section in Figure 3.4(b) tells us that there is an edge on each side on the boundary of the box. In Figure 3.4(c), the same box is illustrated with added Gaussian white noise.

The Canny edge-detector is first applied at different scales with fixed thresholds, $T_{\text{high}} = 0.4$ and $T_{\text{low}} = 0.1$. The output from the edge-detector applied to the noise-free image is presented in Figure 3.5. This figure shows that increasing the scale gives a poorer localization of the boundary of the box.

The output from the edge-detector applied to the noisy image is illustrated in Figure 3.6. Here we see that increasing the scale variable will remove high frequency oscillations, represented as noise.

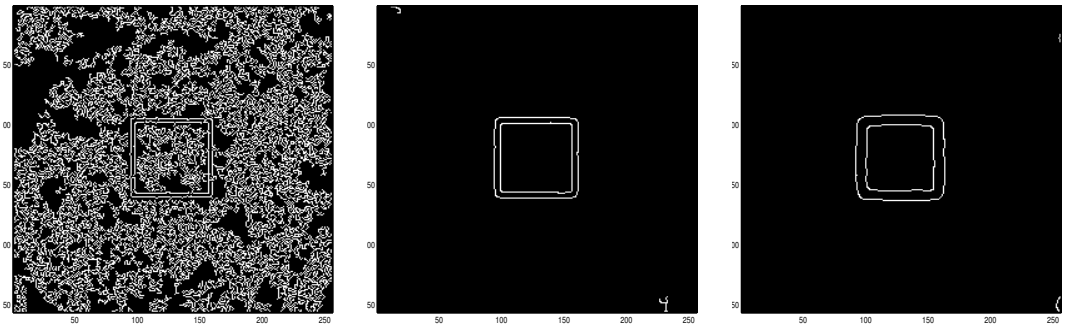
Instead of varying the scale, one can fix the scale s and vary the thresholds. The output with $s = 2^{-1} = 0.5$ and different thresholds is illustrated in Figure 3.7.

This example illustrates why the presence of noise makes it more difficult to detect the correct edges. To predetermine the values of s, T_{high} and T_{low} when noise is present may be difficult. Therefore some experimentation with different values of these parameters is often necessary to obtain a satisfactory result.



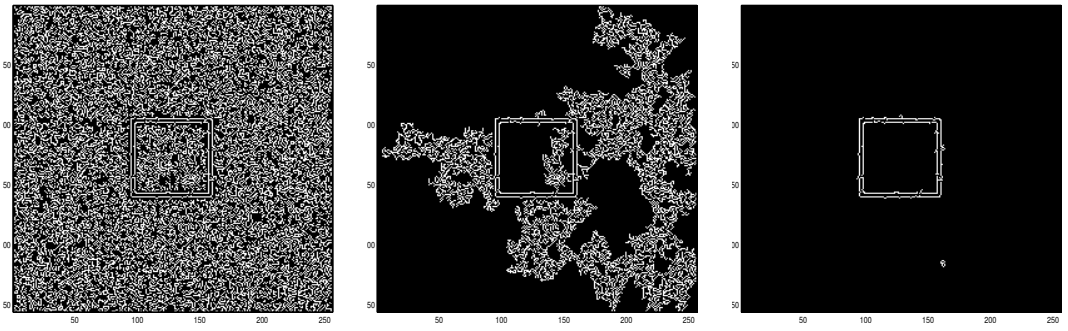
(a) $s = 2^{-1} = 0.5$. (b) $s = 2^{1} = 1$. (c) $s = 2^{2} = 4$.

FIGURE 3.5: Output from edge-detector applied to the noise-free image, computed with different scales and fixed thresholds, $T_{\text{high}} = 0.4$ and $T_{\text{low}} = 0.1$.



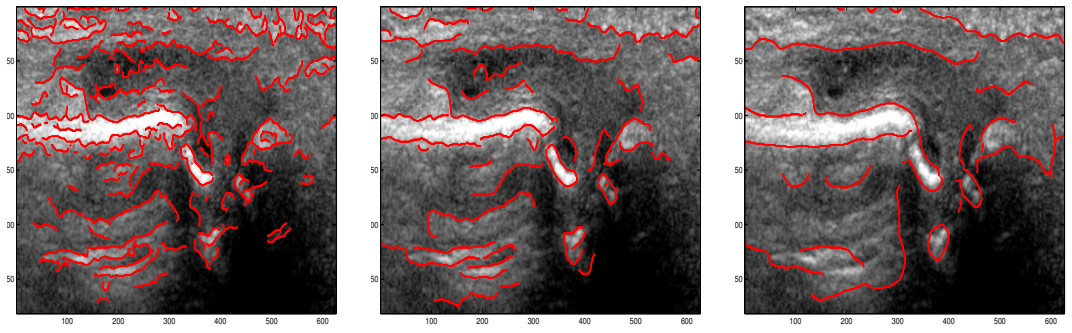
(a) $s = 2^{-1} = 0.5$. (b) $s = 2^{1} = 2$. (c) $s = 2^{2} = 4$.

FIGURE 3.6: Output from edge-detector applied to image with Gaussian white noise, computed with different scales and fixed thresholds, $T_{\text{high}} = 0.4$ and $T_{\text{low}} = 0.1$.



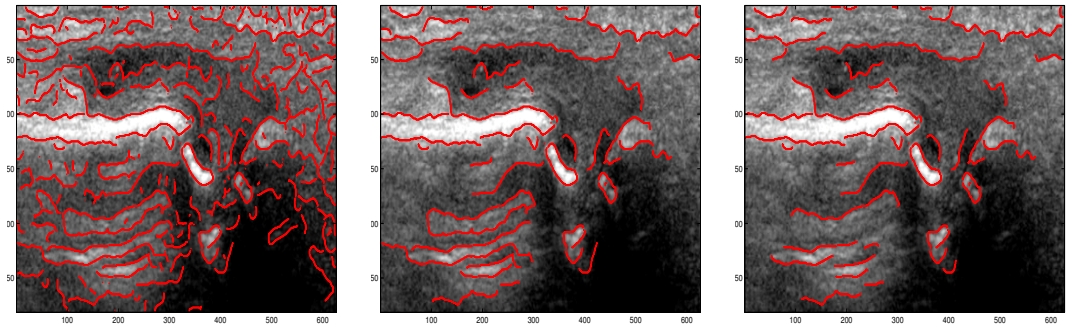
(a) $T_{\text{high}} = 0.2$, $T_{\text{low}} = 0.1$. (b) $T_{\text{high}} = 0.6$, $T_{\text{low}} = 0.1$. (c) $T_{\text{high}} = 0.6$, $T_{\text{low}} = 0.3$.

FIGURE 3.7: Output from edge-detector applied to image with Gaussian white noise, computed for different thresholds and fixed scale, $s = 2^{-1} = 0.5$.



(a) $s = 2$. (b) $s = 4$. (c) $s = 8$.

FIGURE 3.8: Canny edge-detector with with different values of s, and fixed thresholds, $T_{\text{low}} = 0.1$ and $T_{\text{high}} = 0.25$.



(a) $T_{\text{low}} = 0.05$ and $T_{\text{high}} = 0.1$. (b) $T_{\text{low}} = 0.05$ and $T_{\text{high}} = 0.25$. (c) $T_{\text{low}} = 0.15$ and $T_{\text{high}} = 0.25$.

FIGURE 3.9: Canny edge-detector for various thresholds, and fixed scale, $s = 4$.

In Figure 3.8 and 3.9 we see the edge-detector applied to an US image of synovitis for different values of s, T_{high} and T_{low}. We see that there are some difficulties in detection of the correct edges. In Section 5.2.1 we will discuss this problem, and how this problem is solved in this thesis.

Once the boundary has been identified, we can estimate the area of inflammation by using Greens formula

$$\text{Area} = \int_D dA = \frac{1}{2} \int_C x dy - y dx. \tag{3.31}$$

Here D denotes the inflamed region, and C the enclosing boundary curve. Formally we will define the area of inflammation-operator $\mathcal{A} : L^2(\mathbb{R}^2) \to \mathbb{R}$, which takes US images of finger-joints as input, and outputs the area of the inflammed region. In Chapter 5 we will describe how this can be done numerically.

3.3 Scattering wavelets

The two-dimensional wavelet transform used in the Canny edge-detector is computed by rotating a single wavelet along two directions. In this section we will define scattering wavelets which are constructed in the same way. There will however be a slight change in the notation in order to be consistent with the notation used in [Mal12].

Let $\psi \in L^2(\mathbb{R}^2)$ be a wavelet. The wavelet transform will be constructed by dilating the wavelet by dyadic scales, 2^j with $j \in \mathbb{Z}$. The wavelet is also rotated by $r \in G$, where G is the rotation group given by (3.13). The following notation will be used for a dilated and rotated wavelet,

$$\psi_\lambda(x) = 2^{-2j}\psi(2^{-j}r^{-1}x) = 2^{-2j}\psi(\lambda^{-1}x), \quad \lambda = 2^j r \in 2^{\mathbb{Z}} \times G. \tag{3.32}$$

Here the normalization is chosen with respect to the L^1-norm, that is $\|\psi_\lambda\|_{L^1} = \|\psi\|_{L^1}$. The Fourier transform of a dilated and rotated wavelet is

$$\widehat{\psi_\lambda}(\omega) = \widehat{\psi_{2^j r}}(\omega) = \int_{\mathbb{R}^2} \psi(y) e^{-i\langle 2^j r^{-1}\omega, y\rangle} dy = \widehat{\psi}(2^j r^{-1}\omega). \tag{3.33}$$

The windowed scattering transform which will be defined in Chapter 4, can be defined for general wavelets. Complex wavelets that have the following form

$$\psi(x) = e^{i\langle \eta, x\rangle}\theta(x), \tag{3.34}$$

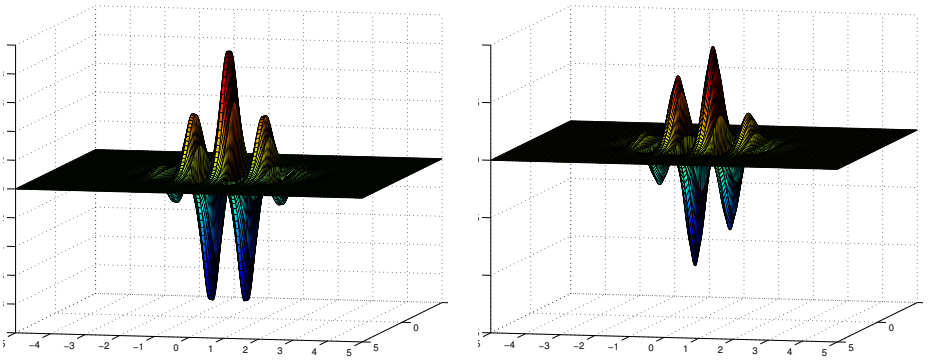
will be of particular interest. This family of wavelets is called Gabor wavelets. Here $\widehat{\theta}(\omega)$ is a real function supported in a neighborhood of $\omega = 0$, and which vanishes near zero. This means that ψ has a real and imaginary part which oscillates like a cosine and a sine respectively. Its Fourier transform $\widehat{\psi} = \widehat{\theta}(\omega - \eta)$, is real and concentrated near η.

Example 3.5. An example of a complex wavelet is the Morlet wavelet which has the following form [BM12]

$$\psi(x) = \alpha e^{-\frac{|x|^2}{2\sigma^2}} \left(e^{i\langle x, \xi\rangle} - \beta \right). \tag{3.35}$$

Here σ determines the spread of the Gaussian envelope, ξ determines the oscillation, α is a normalization constant, and $\beta \ll 1$ is usually adjusted so that the wavelet has zero average. The Fourier transform of the Morlet wavelet is

$$\begin{aligned}
\widehat{\psi}(\omega) &= \frac{\alpha}{2\pi} \int_{\mathbb{R}^2} e^{-\frac{|x|^2}{2\sigma^2}} e^{-i\langle x, \omega - \xi\rangle} dx - \frac{\alpha\beta}{2\pi} \int_{\mathbb{R}^2} e^{-\frac{|x|^2}{2\sigma^2}} e^{-i\langle x, \omega\rangle} dx \\
&= \alpha \left(\widehat{e^{-\frac{|x|^2}{2\sigma^2}}}(\omega - \xi) - \beta \widehat{e^{-\frac{|x|^2}{2\sigma^2}}}(\omega) \right) \\
&= \sigma^2 \alpha \left(e^{-\frac{\sigma^2|\omega - \xi|^2}{2}} - \beta e^{-\frac{\sigma^2|\omega|^2}{2}} \right).
\end{aligned}$$



(a) $\operatorname{Re}\psi$. (b) $\operatorname{Im}\psi$.

FIGURE 3.10: Real and imaginary parts of the Morlet wavelet (3.35).

By choosing $\beta = e^{-(\sigma^2|\xi|^2)/2}$ we see that

$$\widehat{\psi}(\omega) = \sigma^2 \alpha e^{-\frac{\sigma^2(|\omega|^2+|\xi|^2)}{2}} \left(e^{\sigma^2\langle x,\omega\rangle} - 1\right),$$

so that $\widehat{\psi}(0) = \int_{\mathbb{R}^2} \psi(x)dx = 0$. If $\beta \ll 1$ then

$$\widehat{\psi}(\omega) \approx \widehat{\theta}(\omega - \xi), \quad \theta(x) = \alpha e^{-\frac{|x|^2}{2\sigma^2}}.$$

In that case, since

$$\|\theta\|_{L^2}^2 = \alpha^2 \int_{\mathbb{R}^2} e^{-\frac{|x|^2}{\sigma^2}} dx = \alpha^2 \pi \sigma^2,$$

we see that $\|\psi\|_{L^2} = 1$ if we choose $\alpha = \pi^{-1/2}\sigma^{-1}$. The real and imaginary parts of the Morlet wavelet are illustrated in Figure 3.10(a) and 3.10(b). Here $|\xi| = 3$ and $\sigma = 0.85$ in which case $\beta \approx 0.02$.

The corresponding wavelet transform is written as a convolution with the wavelet, which is different from the wavelet transform defined in Definition 3.5. In [Mal12], Mallat refer to this transform as a Littlewood-Paley wavelet transform.

3.3.1 Littlewood-Paley Wavelet Transform

Definition 3.11. *[Mal12] Let G be a finite rotation group and let $\lambda = 2^j r \in 2^{\mathbb{Z}} \times G$. The Littlewood-Paley wavelet transform at scale 2^j and orientation $r \in G$ is given by*

$$W[\lambda]f(x) = f * \psi_\lambda(x) = \int f(u)\psi_\lambda(x-u)du = \int f(u)2^{-2j}\psi(2^{-j}r^{-1}(x-u))du. \tag{3.36}$$

If $f \in L^2(\mathbb{R}^2)$, then $\|W[\lambda]f\|_{L^2} \leq \|f\|_{L^2}\|\psi_\lambda\|_{L^1} = \|f\|_{L^2}\|\psi\|_{L^1}$, so that $W[\lambda]f \in L^2(\mathbb{R}^2)$. Since this wavelet transform is just a convolution, the Fourier transform

of $W[\lambda]f$ is just the product of the Fourier transforms of f and ψ_λ,

$$\widehat{W[\lambda]f}(\omega) = \widehat{f}(\omega)\widehat{\psi}_\lambda(\omega).$$

A wavelet transform at a scale $|\lambda| = 2^J$ only uses wavelets with frequencies $2^j > 2^{-J}$, and hence the information about the signal at low frequencies is lost. To recover these low frequencies, an averaging is done over a spatial domain proportional to 2^J. Let ϕ be a real, symmetric and positive function which is concentrated near zero in both time and frequency. An example is the Gaussian function. The dilated function $\phi_{2^J} = 2^{-2J}\phi(2^{-J}x)$ is then supported on a domain of size proportional to 2^J. We define the averaging operator at scale 2^J as

$$A_J f = f * \phi_{2^J}. \tag{3.37}$$

Note that $\|A_J f\|_{L^2} \leq \|f\|_{L^2}\|\phi_{2^J}\|_{L^1} = \|f\|_{L^2}\|\phi\|_{L^1}$, so that $A_J f \in L^2(\mathbb{R}^2)$. The function ϕ is also known as the scaling function.

Remark 3.12. In the following we will assume that:

- The wavelet ψ satisfies the admissibility condition for a two-dimensional wavelet given in Definition (3.6). This condition implies that $\widehat{\psi}(\omega) = \mathcal{O}(|\omega|)$, and $\widehat{\psi}(0) = 0$.
- $\widehat{\psi}(\omega)$ is real.
- $\widehat{\phi}(\omega)$ is real, symmetric, positive and satisfies $\widehat{\phi}(0) = 1$.
- ϕ and its first partial derivatives are in $L^1(\mathbb{R}^2)$. Since

$$\widehat{\phi}(\omega) = \frac{1}{2\pi}\int \frac{d}{dx_k}\left(\frac{1}{-i\omega_i}e^{-i\langle x,\omega\rangle}\right)\phi(x)dx = \frac{1}{-2\pi i\omega_k}\int e^{-i\langle x,\omega\rangle}\frac{d}{dx_k}\left(\phi(x)\right)dx,$$

  this implies that $|\omega||\widehat{\phi}(\omega)|$ is bounded.

As images are real signals, we will only consider the case when f is real. In that case it is easy to see that $\widehat{f}(-\omega) = \widehat{f^*}(\omega)$. Since $\widehat{\psi}$ is real, we see that $W[-\lambda]f(x) = W[\lambda]f^*(x) = W[\lambda]f(x)$, hence the two rotations r and $-r$ are equivalent. Therefore we only need to consider positive rotations. Let G^+ denote the quotient of G with $\{-\mathbf{1}, \mathbf{1}\}$, where the two rotations r and $-r$ are equivalent. Then we will only consider the case when $\lambda \in 2^{\mathbb{Z}} \times G^+$.

Altogether the wavelet transform at scale 2^J consists of the following set of functions:

$$W_J f = \left\{A_J f, (W[\lambda]f)_{\lambda\in\Lambda_J}\right\}, \tag{3.38}$$

where $\Lambda_J = \left\{\lambda = 2^j r : r \in G^+, 2^j < 2^J\right\}$. We will refer to this set of functions as the filter bank. For $J = \infty$ the filter bank only consists of wavelet transforms and no averaging, $W_\infty f = \{W[\lambda]f\}_{\lambda\in\Lambda_\infty}$ with $\Lambda_\infty = 2^{\mathbb{Z}} \times G^+$. The operator W_J is thus an operator from $L^2(\mathbb{R}^2)$ to the product space generated by copies of $L^2(\mathbb{R}^2)$, with norm given by

$$\|W_J f\|^2 = \|A_J f\|_{L^2}^2 + \sum_{\lambda\in\Lambda_J}\|W[\lambda]f\|_{L^2}^2. \tag{3.39}$$

If $J = \infty$ then

$$\|W_J f\|^2 = \|W_\infty f\|^2 = \sum_{\lambda \in \Lambda_\infty} \|W[\lambda] f\|_{L^2}^2. \tag{3.40}$$

In many applications the wavelet is known as a band-pass filter, whereas the scaling function is referred to as a low-pass filter.

A low-pass filter keeps the low frequencies in the signal and suppresses the higher frequencies. Filtering an image with a low-pass filter will highlight the coarse structure in the image.

A band-pass filter only keeps frequencies that are within a certain range or band. Filtering an image with a band-pass filter will highlight details in the image depending on the frequency band. By dilating and translating the wavelet one can highlight the structure in various scales. A summary of the filter operators is provided in Table 3.1.

Operator	Norm
$W[\lambda] f = f * \psi_\lambda$	$\|W[\lambda] f\|_{L^2} = \|f * \psi_\lambda\|_{L^2}$
$A_J f = f * \phi_{2^J}$	$\|A_J f\|_{L^2} = \|f * \phi_{2^J}\|_{L^2}$
$W_J f = \left\{A_J f, (W[\lambda] f)_{\lambda \in \Lambda_J}\right\}$	$\|W_J f\|^2 = \|A_J f\|_{L^2}^2 + \sum_{\lambda \in \Lambda_J} \|W[\lambda] f\|_{L^2}^2$

TABLE 3.1: Filter operators used for scattering wavelets, and the corresponding norm.

The next proposition gives a condition for $W_J f$ to be a stable representation.

Proposition 3.13. *If there exists $\epsilon > 0$ such that for almost all $\omega \in \mathbb{R}^2$ and all $J \in \mathbb{Z}$*

$$1 - \epsilon \leq |\widehat{\phi}(2^J \omega)|^2 + \sum_{j<J} \sum_{r \in G^+} |\widehat{\psi}(2^j r \omega)|^2 \leq 1, \tag{3.41}$$

then

$$(1 - \epsilon)\|f\|_{L^2}^2 \leq \|W_J f\|^2 \leq \|f\|_{L^2}^2, \quad \forall f \in L^2(\mathbb{R}^2). \tag{3.42}$$

Proof. The Plancherel formula implies that

$$\|W_J f\| = \|\widehat{W_J f}\|^2 = \|\widehat{A_J f}\|_{L^2}^2 + \sum_{\lambda \in \Lambda_J} \|\widehat{W[\lambda] f}\|_{L^2}^2.$$

Using Theorem A.6 we have

$$\|W_J f\|^2 = \int |\widehat{f}(\omega)|^2 \left(|\widehat{\phi}(2^J \omega)|^2 + \sum_{j<J} \sum_{r \in G^+} |\widehat{\psi}(2^j r \omega)|^2 \right) d\omega.$$

Inserting the bounds in (3.41) yields (3.42). □

If $\epsilon < 1$, the operator W_J is non-expansive, invertible and has a stable inverse.

Example 3.6. The Morlet wavelet in Figure 3.10, together with the Gaussian scaling function

$$\phi(x) = \frac{1}{2\pi\sigma^2} e^{-\frac{|x|^2}{2\sigma^2}}, \quad \sigma = 0.7, \tag{3.43}$$

satisfies (3.42) with $\epsilon = 0.25$.

If $\epsilon = 0$, then W_J preserves the Euclidean norm. In this case W_J is a unitary operator.

Example 3.7. The Shannon wavelet is an example where W_J is unitary. Define the one-dimensional wavelet ψ and scaling function ϕ via its Fourier transform

$$\widehat{\psi}(\omega) = e^{i\omega} \chi_{[-1,-1/2]\cup[1/2,1]}(\omega), \tag{3.44}$$

$$\widehat{\phi}(\omega) = \chi_{[-1/2,1/2]}(\omega). \tag{3.45}$$

Here χ_A denotes the characteristic function of the set A. Next, define the two-dimensional wavelets Ψ^k and scaling function Φ as the separable product of $\widehat{\psi}$ and $\widehat{\phi}$,

$$\begin{aligned}
\widehat{\Psi^1}(\omega_1, \omega_2) &= \widehat{\phi}(\omega_1)\widehat{\psi}(\omega_2), \\
\widehat{\Psi^2}(\omega_1, \omega_2) &= \widehat{\psi}(\omega_1)\widehat{\phi}(\omega_2), \\
\widehat{\Psi^3}(\omega_1, \omega_2) &= \widehat{\psi}(\omega_1)\widehat{\psi}(\omega_2), \\
\widehat{\Phi^1}(\omega_1, \omega_2) &= \widehat{\phi}(\omega_1)\widehat{\phi}(\omega_2).
\end{aligned}$$

Then $\Psi = (\Psi^i)_{i=1,2,3}$ will be a collection of two-dimensional wavelets and Φ the two-dimensional scaling function. Let $\omega \in \mathbb{R}^2$ and $J \in \mathbb{Z}$. The support of $\widehat{\Psi}^i(2^j\omega)$ for $j < J$ and $i = 1, 2, 3$ together with the support of $\widehat{\Phi}(2^J\omega)$ define a partition of non-overlapping squares in $\mathbb{R}^2$. The boundaries of these squares define a set of measure zero. Therefore almost every ω will be in the support of exactly one of these squares. Hence

$$|\widehat{\phi}(2^J\omega)|^2 + \sum_{j<J} \sum_{i=1}^{3} |\widehat{\Psi^i}(2^j\omega)|^2 = 1,$$

for almost every $\omega \in \mathbb{R}^2$.

The Shannon wavelet is perfectly localized in the frequency domain, and therefore has poor localization in the time domain. Since the Shannon wavelet preserves the energy of the signal, it will be used when studying energy conservation. For classification we will use the Morlet wavelet which has good localization in both time and frequency.

Definition 3.14. *A scattering wavelet is said to be admissible if there exist* $\eta \in \mathbb{R}^2$ *and* $\rho \geq 0$ *with* $|\widehat{\rho}(\omega)| \leq |\widehat{\phi}(2\omega)|$ *and* $\widehat{\rho}(0) = 1$*, such that the function*

$$\widehat{\Psi}(\omega) = |\widehat{\rho}(\omega - \eta)|^2 - \sum_{k=1}^{\infty} k \left(1 - |\widehat{\rho}(2^{-k}(\omega - \eta))|^2\right), \tag{3.46}$$

satisfies

$$\alpha = \inf_{1 \leq |\omega| \leq 2} \sum_{j=-\infty}^{\infty} \sum_{r \in G} \widehat{\Psi}(2^{-j} r^{-1} \omega) |\widehat{\psi}(2^{-j} r^{-1} \omega)|^2 > 0. \tag{3.47}$$

This condition may look complicated, but it will ensure that the windowed scattering transform, which will be construction in the next chapter, is stable to deformations and noise and is locally translation invariant.

Chapter 4

Scattering

We want to classify images, and need an appropriate metric to measure distances between them. Within each class there is variability due to translations, small deformations and noise. This variability needs to be taken into account, when measuring the distance. Such a metric may be constructed by considering an operator,

$$\Phi : L^2(\mathbb{R}^2) \to \mathcal{H}, \tag{4.1}$$

for a Hilbert space $\mathcal{H}$. If Φ is invariant to translation, stable to deformations and stable to additive noise, then the induced metric on the space $\mathcal{H}$,

$$\forall f, g \in L^2(\mathbb{R}^2), \quad d(f,g) = \|\Phi(f) - \Phi(g)\|_{\mathcal{H}},$$

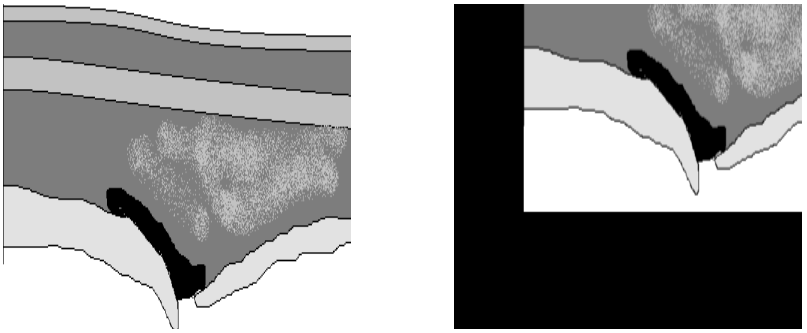
will reduce intra-class variability.

The construction of the operator Φ is based on cascading wavelet transforms with a non-linear modulus operator, and a filtering with a low-pass filter ϕ. The result yields a representation which is locally invariant to translation, stable to deformations and stable to additive noise.

The first section examines translation, deformation and additive noise. We will see that the Fourier modulus is translation invariant and stable to additive noise, but fails to be stable to deformations at high frequencies. A transformation with wider support at high frequencies is needed, which leads us naturally to wavelets. These wavelets will serve as building blocks for the operator Φ, which will be referred to as the windowed scattering transform. The construction of the windowed scattering transform is the content of Section 4.2. In the last section we present the scattering metric, and prove that it has the desired properties outlined in Section 4.1.

4.1 Properties of the representation

In Chapter 2, we mentioned that the human brain is able to recognize objects despite transformations due to translations, noise and small deformations. In this section we will explore these transformations in more detail.



(a) Original image. (b) Translated image.

FIGURE 4.1: Illustration of translation of an image.

4.1.1 Translation invariance

For $c \in \mathbb{R}^2$, the translation operator $T_c : L^2(\mathbb{R}^2) \to L^2(\mathbb{R}^2)$ takes a signal f and translates it, $T_c f(x) = f(x - c)$. In Figure 4.1 an example of a translation of an image is provided. An operator $\Phi : L^2(\mathbb{R}^2) \to \mathcal{H}$ is invariant to translation if

$$\forall c \in \mathbb{R}^2, \quad \|\Phi(T_c f) - \Phi(f)\|_{\mathcal{H}} = 0. \tag{4.2}$$

One such operator is the Fourier modulus. The Fourier transform of a translated signal is

$$\widehat{T_c f}(\omega) = M_c \widehat{f}(\omega),$$

where M_c is the modulation operator, $M_c \widehat{f}(\omega) = e^{-i\langle c,\omega\rangle} \widehat{f}(\omega)$. Taking the modulus will remove the complex phase, so that $|\widehat{T_c f}(\omega)| = |e^{-i\langle c,\omega\rangle} \widehat{f}(\omega)| = |\widehat{f}(\omega)|$.

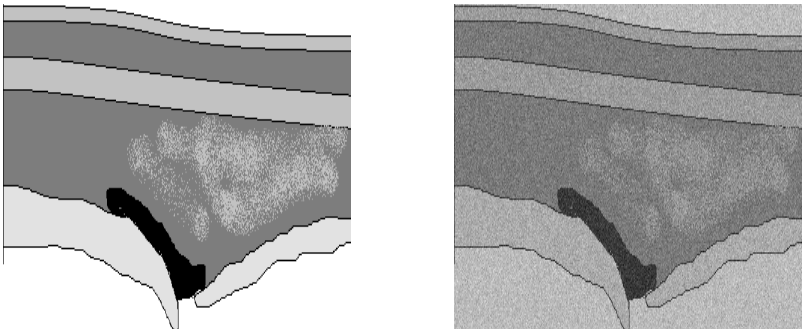
Later we will define the windowed scattering transform at a scale 2^J. This transformation will not by fully translation invariant, but locally translation invariant. An operator Φ is locally translation invariant at scale 2^J if there exists a constant $C > 0$ such that for all $c \in \mathbb{R}^2$,

$$\|\Phi(T_c f) - \Phi(f)\|_{\mathcal{H}} \leq C 2^{-J} |c| \|f\|. \tag{4.3}$$

We are mainly interested in the case when f is a compactly supported function. The amount of translation is therefore limited by the support of f. In this case one may choose a scale, so that the windowed scattering transform will behave almost as a fully translation invariant operator.

4.1.2 Stability to additive noise

A common problem in US images is the presence of noise. Additive noise can be described as a random process that is added to the signal. Let $f_0(x)$ denote the



(a) Noise-free image. (b) Image with added noise.

FIGURE 4.2: Illustration of an image with and without additive noise.

noise-free image, and let $n(x)$ denote the additive noise. Then

$$f(x) = f_0(x) + n(x) \tag{4.4}$$

is the image with added noise. Figure 4.2 shows an example of an image with, and without additive noise.

To ensure stability to additive noise we impose a Lipschitz continuity condition on Φ. This means that there exists a constant $C > 0$ such that for all $f, f' \in L^2(\mathbb{R}^2)$,

$$\|\Phi(f') - \Phi(f)\|_{\mathcal{H}} \leq C\|f' - f\|_{L^2}. \tag{4.5}$$

This implies that

$$\|\Phi(f) - \Phi(f_0)\|_{\mathcal{H}} \leq C\|f - f_0\|_{L^2} = C\|n\|_{L^2}.$$

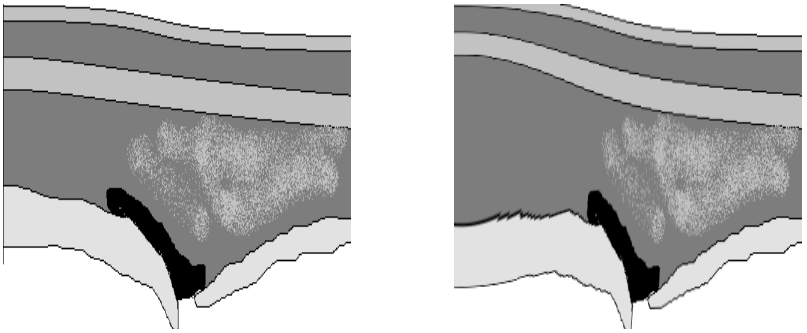
The difference between the representation of the noise-free signal and the representation of the signal with noise is thus bounded by the L^2-norm of the added noise.

If Φ is the Fourier modulus, then the Parseval equality implies that Φ is Lipschitz continuous with $C = 1$.

4.1.3 Stability to deformations

By deformations we mean C^2 diffeomorphisms on $\mathbb{R}^2$. The space of such functions will be referred to as $C^2(\mathbb{R}^2)$. Let $X, Y \subseteq \mathbb{R}^2$. A function $\tau : X \to Y$ is a C^2 diffeomorphism if it is two times continuously differentiable, bijective, and its inverse $\tau^{-1} : Y \to X$, is also two times continuously differentiable. An example is a dilation $\tau(x) = \epsilon x$ with $\epsilon \in \mathbb{R}^2 \setminus \{(0,0)\}$. Figure 4.3 illustrates two images which can be obtained from each other via a diffeomorphism.

A representation is stable to deformations if it is Lipschitz continuous to the action of C^2 diffeomorphisms. A deformation of a signal $f \in L^2(\mathbb{R}^2)$, supported in $\Omega \subseteq \mathbb{R}^2$, may be written as $T_\tau f(x) = f(x - \tau(x))$. This is similar to a translation, but the amount of translation is now dependent on the position in the image.



(a) Original image. (b) Deformed image

FIGURE 4.3: Illustration of a deformation of an image.

The size of the deformation is determined by the maximum amplitude of the diffeomorphism, $\sup_{x\in\Omega}|\tau(x)|$, the norm of the gradient, $\sup_{x\in\Omega}|\nabla\tau(x)|$, and the Hessian tensor, $\sup_{x\in\Omega}|H\tau(x)|$. The amount of deformation over a compact set $\Omega \subseteq \mathbb{R}^2$ is measured by the norm

$$\|\tau\|_\infty = \sup_{x\in\Omega}|\tau(x)| + \sup_{x\in\Omega}|\nabla\tau(x)| + \sup_{x\in\Omega}|H\tau(x)|.$$

As mentioned in Section 4.1.1 we will construct an operator which is locally translation invariant. Our representation $\Phi(f)$ is therefore stable to deformations and locally translation invariant at scale 2^J if for any compact set $\Omega \subseteq \mathbb{R}^2$ there exists a constant $C > 0$ such that for all $f \in L^2(\mathbb{R}^2)$ supported in Ω and all $\tau \in C^2(\mathbb{R}^2)$,

$$\|\Phi(f) - \Phi(T_\tau f)\| \leq C\|f\|_{L^2}\left(2^{-J}\sup_{x\in\Omega}|\tau(x)| + \sup_{x\in\Omega}|\nabla\tau(x)| + \sup_{x\in\Omega}|H\tau(x)|\right). \quad (4.6)$$

By adjusting the scale, one can control the amount of translation invariance so that the condition (4.6) becomes

$$\|\Phi(f) - \Phi(T_\tau f)\| \leq C\|f\|_{L^2}\left(\sup_{x}|\nabla\tau(x)| + \sup_{x\in\Omega}|H\tau(x)|\right). \quad (4.7)$$

Now the following question arises, "how do we find the operator Φ?". The natural thing to do at this stage is to check if the Fourier modulus is stable to deformations. If it was, then it would serve as an excellent candidate for Φ. Unfortunately, the next example shows that this is not the case.

Example 4.1. Let $f(x) = e^{i\langle\xi,x\rangle}\theta(x)$, where θ is regular, has fast decay, and $\widehat{\theta}(\omega)$ is concentrated near the origin. Let $\tau(x) = \epsilon x$ be a dilation. The deformed signal will then be $T_\tau f(x) = f(x - \tau(x)) = f((1-\epsilon)x)$. Taking the Fourier transform gives us

$$\widehat{f((1-\epsilon)x)} = \int \theta((1-\epsilon)x)e^{i\xi(1-\epsilon)x}e^{-i\omega x}dx = \frac{1}{1-\epsilon}\widehat{\theta}\left(\frac{\omega-(1-\epsilon)\xi}{1-\epsilon}\right).$$

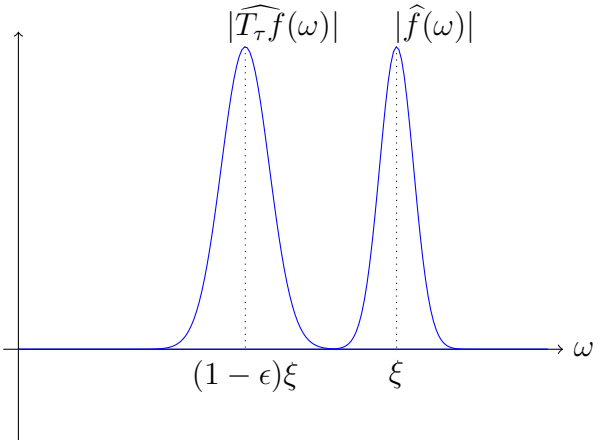


FIGURE 4.4: Showing instability to the action of diffeomorphisms for the Fourier modulus.

The central frequency has now been shifted from ξ to $(1-\epsilon)\xi$. Since $\widehat{f}(\omega) = \widehat{\theta}(\omega - \xi)$ we can choose ξ large enough so that when $\xi - (1-\epsilon)\xi = \epsilon\xi$ is sufficiently large, the difference $\| |\widehat{f}(\omega)| - |\widehat{T_\tau f}(\omega)| \|$ is non-negligible, see Figure 4.4. In fact $\| |\widehat{f}(\omega)| - |\widehat{T_\tau f}(\omega)| \| \sim |\epsilon| \|f\| |\xi|$, so that given any constant C there exists $\xi \in \mathbb{R}^2$ so that $\| |\widehat{f}(\omega)| - |\widehat{T_\tau f}(\omega)| \| > C|\epsilon| \|f\|$. Hence the Fourier modulus is not stable to deformations at high frequencies.

To remedy this problem, a transformation with wider support in the Fourier domain at high frequencies is needed. This is achieved by a wavelet transform.

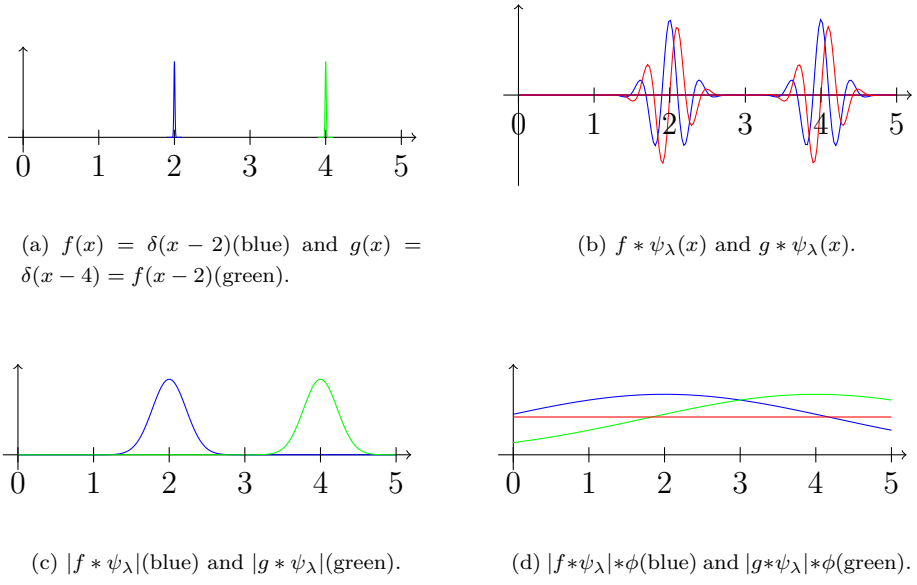
4.1.4 Building invariant structure

We want to build a representation Φ which is invariant to translations, stable to additive noise and stable to deformations. In section 4.1.3 we saw that the Fourier modulus failed to be stable to deformations. The wavelet transform defined in (3.36) is stable to deformations, but fails to be translation invariant. In fact, it is covariant to translations, meaning that translating the signal will also translate the wavelet transform:

$$\begin{aligned} W[\lambda]T_c f(x) &= \int T_c f(u)\psi_\lambda(x-u)du = \int f(u-c)\psi_\lambda(x-u)du \\ &= \int f(u)\psi_\lambda(x-c-u)du = W[\lambda]f(x-c). \end{aligned}$$

Example 4.2. Let $f(x) = \delta(x-2)$ and let $g(x) = \delta(x-4) = f(x-2)$ be two signals which can be obtained from each other by a translation. Figure 4.5(a) illustrates the two signals. Applying the wavelet transform (3.36) to these signals will give back the wavelet centred around $x = 2$ and $x = 4$, see Figure 4.5(b). This illustrates the fact that the wavelet transform is not translation invariant.

To get a translation invariant representation one could try to apply the modulus operator as we did with the Fourier transform. This will remove the complex phase and return the envelope, see Figure 4.5(c). The envelope is more regular, but not translation invariant. To obtain a translation invariant representation one may



(a) $f(x) = \delta(x-2)$(blue) and $g(x) = \delta(x-4) = f(x-2)$(green).

(b) $f * \psi_\lambda(x)$ and $g * \psi_\lambda(x)$.

(c) $|f * \psi_\lambda|$(blue) and $|g * \psi_\lambda|$(green).

(d) $|f*\psi_\lambda|*\phi$(blue) and $|g*\psi_\lambda|*\phi$(green).

FIGURE 4.5: A wavelet modulus with averaging applied to $f(x) = \delta(x-2)$ and $g(x) = \delta(x-4) = f(x-2)$.

average over the whole domain. This will give you the same constant for both f and g, hence a translation invariant representation. This representation is visualized as a red line in Figure 4.5(d). However, one loses a lot of information about the signals. Instead one can do an averaging over a smaller domain by filtering with a low-pass filter ϕ, see the blue and green lines in Figure 4.5(d). This will not give you a translation invariant representation. However if we only consider a finite interval, say the interval $[1, 5]$, then the difference between the two functions f and g will be much smaller than before applying any filtering. Moreover, the difference between the two functions will be smaller if the translation is smaller. Hence we have an example of a locally translation invariant representation. This kind of representation is called a scale-invariant feature transform (SIFT).

From an operator which commutes with the translation operator one can construct a translation invariant operator by taking the average. If M is an operator such that the commutator $[M, T_c] = MT_c - T_cM = 0$, then $\int Mf(x)dx$ is translation invariant. The wavelet transform is indeed commuting with the translation operator. However, since our wavelets are assumed to have zero average we see that for any $\lambda \in \Lambda_\infty$,

$$\begin{aligned}\int W[\lambda]f(x)dx &= \int\int f(u)\psi_\lambda(x-u)dudx \\ &= \int f(u)du \int \psi_\lambda(x)dx = \int f(u)du \cdot 0 = 0,\end{aligned}$$

hence all information is lost. It turns out that if M is any linear operator that commutes with the translation operator we have $\int M(W[\lambda]f)(x)dx = 0$ [BM12], hence M needs to be non-linear. Moreover, we need $\int M(W[\lambda]f)(x)dx$ to preserve stability to deformations and additive noise.

This suggests M to be the modulus operator. Hence a translation invariant representation can be obtained by first taking the modulus of the wavelet transform, and then doing the averaging,

$$\int |f * \psi_\lambda(x)| dx = \|f * \psi_\lambda\|_{L^1(\mathbb{R}^2)}.$$

A locally translation invariant representation can be obtained by filtering with a low-pass filter ϕ,

$$\int |f * \psi_\lambda(x)| \phi(y - x) dx.$$

Let us illustrate the use of the wavelet transform and modulus with two examples.

Example 4.3. Let $f(x) = e^{i\xi x}$, and suppose $\xi \in \operatorname{supp} \widehat{\psi_\lambda}$, then

$$W[\lambda]f(x) = \int e^{i\xi u} \psi_\lambda(x - u) du = -\int e^{i\xi(x-v)} \psi_\lambda(v) dv = -e^{i\xi x} \widehat{\psi_\lambda}(\xi),$$

where we made the substitution $v = x - u$. If we apply the modulus, we see that we remove the complex phase and are left with the constant $|f * \psi_\lambda(x)| = |-e^{i\xi x} \widehat{\psi_\lambda}(\xi)| = |\widehat{\psi_\lambda}(\xi)|$. If ϕ is a low-pass filter with $\hat{\phi}(0) = 1$, then

$$\int |f * \psi_\lambda(x)| \phi(y - x) dx = |\widehat{\psi_\lambda}(\xi)| \int \phi(x) dx = |\widehat{\psi_\lambda}(\xi)|.$$

As the output is independent of x, this is a translation invariant representation.

Example 4.4. Let $f(x) = e^{i\xi_1 x} + a e^{i\xi_2 x}$ for $x \in \Omega$, where $\Omega \subset \mathbb{R}$ is compact. If $\xi_1, \xi_2 \in \operatorname{supp} \widehat{\psi_\lambda}$, then by similar calculation as in Example 4.3 we have

$$W[\lambda]f(x) = -\left(e^{i\xi_1 x} \widehat{\psi_\lambda}(\xi_1) + a e^{i\xi_2 x} \widehat{\psi_\lambda}(\xi_2)\right) = -e^{i\xi_1 x} \left(\widehat{\psi_\lambda}(\xi_1) + a e^{i(\xi_2 - \xi_1)x} \widehat{\psi_\lambda}(\xi_2)\right).$$

Applying the modulus gives us

$$|f * \psi_\lambda(x)| = \left|\widehat{\psi_\lambda}(\xi_1) + a e^{i(\xi_2 - \xi_1)x} \widehat{\psi_\lambda}(\xi_2)\right|.$$

An illustration of the real part of the original function, and the real part of the wavelet transform is provided in Figure 4.6. The wavelet used in this case is the Morlet wavelet (3.35). To obtain a translation invariant representation we do an averaging over the domain Ω. Let ϕ be a low-pass filter whose support is of the size of Ω. Then

$$\int_\Omega |f * \psi_\lambda(x)| * \phi(x) dx = \int_\Omega \left|\widehat{\psi_\lambda}(\xi_1) + a e^{i(\xi_2 - \xi_1)x} \widehat{\psi_\lambda}(\xi_2)\right| * \phi(x) dx \approx |\widehat{\psi_\lambda}(\xi_1)|,$$

and we see that the averaging removes the high frequency oscillations. To recover these high frequencies we can compute a new wavelet transform, $W[\lambda']|f * \psi_\lambda(x)| =$

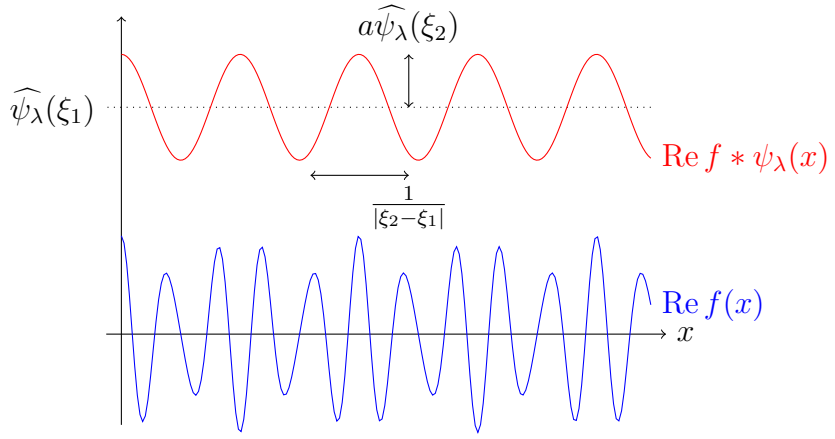


FIGURE 4.6: Showing the real part of the function and wavelet transform in Example 4.4.

.

$|f * \psi_\lambda| * \psi_{\lambda'}(x)$ with $(\xi_2 - \xi_1) \in \operatorname{supp} \psi_{\lambda'}$. This yields

$$\begin{aligned} |f * \psi_\lambda| * \psi_{\lambda'}(x) &= \int \left|\widehat{\psi_\lambda}(\xi_1) + ae^{i(\xi_2-\xi_1)u}\widehat{\psi_\lambda}(\xi_2)\right| \psi_{\lambda'}(x-u)du \\ &\overset{\widehat{\psi}(0)=0}{=} a\widehat{\psi_\lambda}(\xi_2) \int e^{i(\xi_2-\xi_1)u}\psi_{\lambda'}(x-u)du \\ &= -ae^{i(\xi_2-\xi_1)x}\widehat{\psi_\lambda}(\xi_2)\widehat{\psi_{\lambda'}}(\xi_2-\xi_1). \end{aligned}$$

Applying the modulus gives us the constant

$$||f * \psi_\lambda| * \psi_{\lambda'}(x)| = a|\widehat{\psi_\lambda}(\xi_2)||\widehat{\psi_{\lambda'}}(\xi_2 - \xi_1)|.$$

As in the previous example, filtering a constant with the low-pass filter ϕ will output the same constant.

4.2 The windowed scattering transform

Example 4.4 gives an idea on how to construct a representation invariant to translation, stable to additive noise, and stable to deformations. A translation invariant representation is obtained by applying the wavelet transform to the signal f, taking the modulus and doing an averaging. In doing so, information about the high frequencies are lost. Since the wavelet transform is a redundant representation, one can (for particular wavelets) recover the phase from the modulus [WdM13]. The loss of information is thus due to the averaging. To recover these high frequencies, a new wavelet transform is applied to the signal before the averaging is done.

Definition 4.1. *[Mal12] A sequence $p = (\lambda_1, \lambda_2, \cdots, \lambda_m)$ with $\lambda_k \in \Lambda_\infty = 2^{\mathbb{Z}} \times G^+$ is called a path. The empty path is denoted $p = \emptyset$. For $f \in L^2(\mathbb{R}^2)$, and admissible wavelet ψ (3.47), define*

$$U[\lambda]f = |f * \psi_\lambda| = |W[\lambda]f|. \tag{4.8}$$

A scattering propagator is a path ordered product of non-commutative operators defined by

$$U[p] = U[\lambda_m] \cdots U[\lambda_2] U[\lambda_1], \tag{4.9}$$

with $U[\emptyset] = Id$.

Since $\|U[\lambda]f\|_{L^2} \leq \|\psi_\lambda\|_{L^1} \|f\|_{L^2}$ (A.8), the operator $U[\lambda]$ is well defined and $U[\lambda]f \in L^2$. It follows by the triangle inequality that for any path p of length m we have $\|U[p]f\|_{L^2} \leq \|\psi_\lambda\|_{L^1}^m \|f\|_{L^2}$. Hence $U[p]$ is well defined and $U[p]f \in L^2$.

The path variable p can be viewed as a kind of frequency variable. It can be manipulated in several ways. For example, given two paths p and p', the concatenation is written $p^\frown p' = (\lambda_1, \cdots \lambda_m, \lambda'_1, \cdots, \lambda'_{m'})$, and by definition of the scattering propagator we have $U[p^\frown p'] = U[p']U[p]$. Note that the operation $^\frown$ is not commutative, that is $p^\frown p' \neq p'^\frown p$. One can also scale and rotate a path. Let $2^l r_\theta \in 2^{\mathbb{Z}} \times G^+$ be a scaling by 2^l and a rotation by θ. Then $2^l r_\theta p = (2^l r_\theta \lambda_1, 2^l r_\theta \lambda_2, \cdots, 2^l r_\theta \lambda_m)$. It should be understood that if $2^l r_\theta \lambda_k = 2^{l+l_k} r_{\theta+\theta_k} \notin 2^{\mathbb{Z}} \times G^+$, one should choose the rotation $r_{-(\theta+\theta_k)}$.

In Example 4.4, the scattering propagator was computed along the path $p = (\lambda, \lambda')$ so that,

$$U[p]f = U[\lambda']U[\lambda]f = ||f * \psi_\lambda| * \psi_{\lambda'}|.$$

To obtain a translation invariant representation an averaging was applied. In all practical applications, signals such as audio and images are compactly supported, that is they vanish outside some bounded region. It is therefore not necessary to have a representation which is invariant to all translations, as the largest possible translation is determined by the support of the image. If for example f is an image supported on the square $[0, 2^J] \times [0, 2^J]$, then any translation $T_c f(x) = f(x - c)$ with $|c| > 2^{J+1/2}$ will translate the image outside it's support. In image and audio recognition it is therefore often better to compute locally translation invariant representations. This is obtained by applying a low-pass filter ϕ to the representation, which is scaled according to a predefined scale 2^J, $\phi_{2^J}(x) = 2^{-2J}\phi(2^{-J}x)$. This results in a windowed scattering transform.

Definition 4.2. *[Mal12] For fixed* $J \in \mathbb{Z}$ *let*

$$\mathcal{P}_J = \{p = (\lambda_1, \lambda_2, \cdots, \lambda_m) : \lambda_k \in \Lambda_J, m \in \mathbb{N} \cup \{0\}\}. \tag{4.10}$$

For each $p \in \mathcal{P}_J$ *and* $f \in L^2(\mathbb{R}^2)$ *the windowed scattering transform is defined as*

$$S_J[p]f(x) = U[p]f * \phi_{2^J}(x) = \int U[p]f(u)\phi_{2^J}(x - u)du. \tag{4.11}$$

Since $U[p]f \in L^2(\mathbb{R}^2)$ whenever $f \in L^2(\mathbb{R}^2)$, and $\phi \in L^1(\mathbb{R}^2)$ we see that $S_J[p]f \in L^2(\mathbb{R}^2)$ whenever $f \in L^2(\mathbb{R}^2)$, so that $S_J[p] : L^2(\mathbb{R}^2) \to L^2(\mathbb{R}^2)$. The norm of $S_J[p]f$ is given by

$$\|S_J[p]f\|_{L^2}^2 = \int_{\mathbb{R}^2} |S_J[p]f(x)|^2 dx = \int_{\mathbb{R}^2} ||||\psi_{\lambda_1} * f(x)| * \cdots| * \psi_{\lambda_m}| * \phi_{2^J}(x)|^2 dx.$$

To compute the windowed scattering transform we define the one-step propagator

$$U_J f = \left\{ A_J f, (U[\lambda] f)_{\lambda \in \Lambda_J} \right\}, \tag{4.12}$$

where $A_J f = f * \phi_{2^J}$ and $U[\lambda] f = |f * \psi_\lambda|$. By definition we have $A_J U[p] f = S_J[p] f$, and the concatenation property of the scattering propagator implies that $U[\lambda] U[p] f = U[p^\frown \lambda] f$. Hence

$$U_J U[p] f = \left\{ S_J[p] f, (U[p^\frown \lambda] f)_{\lambda \in \Lambda_J} \right\}. \tag{4.13}$$

Let

$$\Lambda_J^m = \{ p = (\lambda_1, \cdots, \lambda_m) : \lambda_k \in \Lambda_J, k = 1, 2, \cdots, m \} = \underbrace{\Lambda_J \times \Lambda_J \times \cdots \times \Lambda_J}_{m \text{ times}}. \tag{4.14}$$

That is Λ_J^m is the set of all paths of length m where each coordinate λ_k belongs to Λ_J. Then it follows that

$$U_J U[\Lambda_J^m] f = \left\{ S_J[\Lambda_J^m] f, U[\Lambda_J^{m+1}] f \right\}. \tag{4.15}$$

The notation $U[\Lambda_J^m] f$ and $S_J[\Lambda_J^m] f$ means that U and S_J are applied to every path $p \in \Lambda_J^m$ respectively. The output will thus be a vector of functions in $L^2(\mathbb{R}^2)$ of length $|\Lambda_J|^m$. The corresponding norm is given by

$$\begin{aligned} \|S_J[\Lambda_J^m] f\|^2 &= \sum_{p \in \Lambda_J^m} \|S_J[p] f\|_{L^2}^2, \quad \text{and} \\ \|U[\Lambda_J^m] f\|^2 &= \sum_{p \in \Lambda_J^m} \|U[p] f\|_{L^2}^2. \end{aligned}$$

At $m = 0$ we have $\Lambda_J^0 = \{\emptyset\}$, hence $S_J[\emptyset] f = f * \phi_J$. By iteratively applying the one-step propagator U_J to $U[\Lambda_J^m] f$ as in (4.15), the length of each path is extended by one. We say that a new layer is added. Hence m denotes the number of layers. Since

$$\mathcal{P}_J = \bigcup_{m=0}^{\infty} \Lambda_J^m,$$

we see that applying U_J an infinite number of times will give you the windowed scattering transform computed along all possible paths.

Later we will define the scattering metric where we apply the windowed scattering transform to every path $p \in \mathcal{P}_J$. This means that we need to make sense of $S_J[\mathcal{P}_J] f(x)$. This representation can be viewed as the infinite vector

$$S_J[\mathcal{P}_J] f(x) = \begin{pmatrix} f * \phi_{2^J}(x) \\ |f * \psi_{\lambda_1}| * \phi_{2^J}(x) \\ ||f * \psi_{\lambda_1}| * \psi_{\lambda_2}| * \phi_{2^J}(x) \\ \vdots \end{pmatrix}_{\lambda_k \in \Lambda_J}.$$

The corresponding norm is

$$\begin{aligned}\|S_J[\mathcal{P}_J]f\|^2 &= \sum_{p\in\mathcal{P}_J} \|S_J[p]f\|_{L^2}^2 \\ &= \sum_{m=0}^{\infty} \sum_{(\lambda_1,\lambda_2,\cdots,\lambda_m)\in\Lambda_J^m} \|||f*\psi_{\lambda_1}|*\cdots|*\psi_{\lambda_m}|*\phi_{2^J}\|_{L^2}^2.\end{aligned} \tag{4.16}$$

The Hilbert space $\mathcal{H}$ will therefore be the range of this operator. This will be the product space generated by copies of $L^2(\mathbb{R}^2)$.

Remark 4.3. In practice the windowed scattering transform is computed along a finite number of layers. The energy in the deep layers will become negligible. Denote

$$\mathcal{P}_J^M = \bigcup_{m=0}^{M} \Lambda_J^m,$$

the set of all paths of length smaller than or equal to M. If Λ_J consists of K elements, then Λ_J^m will consist of K^m paths. The Hilbert space $\mathcal{H}$ will thus be the product of $N_J = M \times \sum_{m=0}^{M} K^m$ copies of $L^2(\mathbb{R}^2)$, i.e

$$\mathcal{H} = \bigotimes_{k=1}^{N_J} L^2(\mathbb{R}^2).$$

Moreover as digital images have a discrete representation, we can view an image as an element of $\mathrm{M}_{N,K}(\mathbb{R})$, where $\mathrm{M}_{N,K}(\mathbb{R})$ denotes the set of all $N \times K$ matrices over $\mathbb{R}$. Here N and K are the number of pixels in the vertical and horizontal direction respectively. In this case

$$\mathcal{H} = \bigotimes_{k=1}^{N_J} \mathrm{M}_{N,K}(\mathbb{R}) \cong \mathbb{R}^{N_J\times N\times K},$$

where the symbol $\cong$ means that the two spaces are isomorphic. The distance between any two images can then be calculated by the usual Euclidean distance. A more detailed review of the discrete case will be presented in Chapter 5.

In Table 4.1 we summarize the operators used to compute the windowed scattering transform of a function $f \in L^2(\mathbb{R}^2)$.

Computing the windowed scattering transform of a signal $f \in L^2(\mathbb{R}^2)$ can be summarized in the following steps:

1. Choose a wavelet ψ and a corresponding low-pass filter ϕ, which satisfy the admissibility conditions in Definition 3.14, and the conditions given in Remark 3.12.

2. Choose a finite rotation-group G, an appropriate scale 2^J, and a number of layers M.

3. For each $m = 0, 1, \cdots, M$ compute $U_J U[\Lambda_J^m]f$. The output of the windowed scattering transform will be the collection of functions $S_J[\mathcal{P}_J^M]f$. If $M = \infty$, then the output is $S_J[\mathcal{P}_J]f$.

Operator	Norm
$U[\lambda]f = \lvert f * \psi_\lambda \rvert$	$\lVert U[\lambda]f \rVert_{L^2} = \lVert f * \psi_\lambda \rVert_{L^2}$
$U[p]f = \lvert\lvert\lvert f * \psi_{\lambda_1}\rvert \cdots \rvert * \psi_{\lambda_m}\rvert$	$\lVert U[p]f \rVert_{L^2} = \lVert\lvert\lvert f * \psi_{\lambda_1}\rvert \cdots \rvert * \psi_{\lambda_m}\rVert_{L^2}$
$U[\Lambda_J^m]f = (U[p]f)_{p\in\Lambda_J^m}$	$\lVert U[\Lambda_J^m]f \rVert^2 = \sum_{p\in\Lambda_J^m} \lVert U[p]f \rVert_{L^2}^2$
$U_J f = \left\{A_J f, (U[\lambda]f)_{\lambda\in\Lambda_J}\right\}$	$\lVert U_J f \rVert^2 = \lVert A_J f \rVert_{L^2}^2 + \sum_{\lambda\in\Lambda_J} \lVert U[\lambda]f \rVert_{L^2}^2$
$S_J[p]f = U[p]f * \phi_{2^J}$	$\lVert S_J[p]f \rVert_{L^2} = \lVert U[p]f * \phi_{2^J} \rVert_{L^2}$
$S_J[\Lambda_J^m]f = (S_J[p]f)_{p\in\Lambda_J^m}$	$\lVert S_J[\Lambda_J^m]f \rVert^2 = \sum_{p\in\Lambda_J^m} \lVert S_J[p]f \rVert_{L^2}^2$
$S_J[\mathcal{P}_J]f = (S_J[p]f)_{p\in\mathcal{P}_J}$	$\lVert S_J[\mathcal{P}_J]f \rVert^2 = \sum_{p\in\mathcal{P}_J} \lVert S_J[p]f \rVert_{L^2}^2$

TABLE 4.1: Scattering operators, and their corresponding norm.

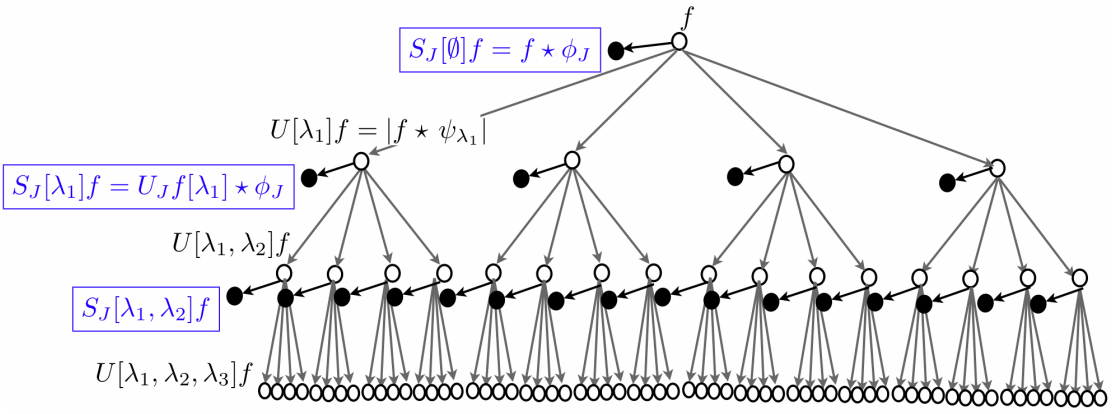


FIGURE 4.7: The scattering propagator U_J applied to each layer.

In Figure 4.7[1] a visualization of scattering propagator applied to the two first layers is provided. In the first iteration the output is simply $S_J[\emptyset]f$. Furthermore, the operator U is applied along all scales and rotations. In this figure, the number of rotations and scales are 4, meaning that there will be 16 different paths to the second layer, and $4^3 = 64$ different paths to the third layer. However, numerical experiments show that for appropriate wavelets, the energy in the signal is mostly concentrated along frequency-decreasing paths [Mal12].

4.2.1 Frequency-decreasing paths

Definition 4.4. *A frequency-decreasing path, is a path* $p = (\lambda_1, \lambda_2, \cdots, \lambda_m)$ *such that* $|\lambda_{k+1}|^{-1} < |\lambda_k|^{-1}$. *The set of all frequency-decreasing paths of length* m

[1]This Figure is taken from [Mal12]. The notation ϕ_J means ϕ_{2^J}.

will be denoted $\Lambda^m_{J\downarrow}$*, and the set of all frequency-decreasing paths will be denoted* $\mathcal{P}_{J\downarrow} = \bigcup_{m=0}^{\infty} \Lambda^m_{J\downarrow}$.

If ψ is a one-dimensional wavelet such that its Fourier transform is supported in the interval $[1/2, 1]$, then $\widehat{\psi}(2^j\omega)$ is supported in the interval $[2^{-(j+1)}, 2^{-j}]$. It follows that $\widehat{\psi}(2^{j+1}\omega)$ is supported in $[2^{-(j+2)}, 2^{-(j+1)}]$. Hence, when increasing the scale, the support of the Fourier transform of the dilated wavelet will be concentrated at lower frequencies. This explains why a path $p = (\lambda_1, \lambda_2, \cdots, \lambda_m)$ with $|\lambda_{k+1}|^{-1} < |\lambda_k|^{-1}$ is called frequency-decreasing. If $|\lambda_{k+1}|^{-1} > |\lambda_k|^{-1}$, the path is called frequency-increasing.

We will prove that for real signals, no energy is captured by frequency-increasing paths when using the Shannon wavelet. Hence it suffices to compute the windowed scattering transform along frequency-decreasing paths. As we are mainly interested in showing that the energy is captured by these paths, we will consider the one-dimensional case.

Proposition 4.5. *Let* ψ *be the Shannon wavelet given in* (3.45)*. If* $f \in L^2(\mathbb{R})$ *is a real valued signal, then* $\||f * \psi_\lambda| * \psi_{\lambda'}\|_{L^2}$ *is non-zero only if* $|\lambda'|^{-1} < |\lambda|^{-1}$.

Proof. The wavelet ψ is given by

$$\widehat{\psi}(\omega) = e^{i\omega}\chi_{[-1,-1/2]\cup[1/2,1]}(\omega) = e^{i\omega}\chi_{[-1,-1/2]}(\omega) + e^{i\omega}\chi_{[1/2,1]}(\omega).$$

If $\lambda = 2^j$, then

$$\widehat{\psi}_\lambda(\omega) = \widehat{\psi}(2^j\omega) = e^{i2^j\omega}\chi_{[-2^{-j},-2^{-(j+1)}]}(\omega) + e^{i2^j\omega}\chi_{[2^{-(j+1)},2^{-j}]}(\omega).$$

Let $f \in L^2(\mathbb{R})$, then

$$\begin{aligned}\widehat{W[\lambda]f}(\omega) = \widehat{f * \psi_\lambda}(\omega) &= \widehat{f}(\omega)\left(e^{i2^j\omega}\chi_{[-2^{-(j)},-2^{-(j+1)}]}(\omega) + e^{i2^j\omega}\chi_{[2^{-(j+1)},2^{-j}]}(\omega)\right)\\ &= \left(\widehat{f}(-\omega)e^{-i2^j\omega} + \widehat{f}(\omega)e^{i2^j\omega}\right)\chi_{[2^{-(j+1)},2^{-j}]}(\omega)\end{aligned}$$

If f is real-valued then $\widehat{f}(-\omega) = \widehat{f}(\omega)^*$. Hence

$$\begin{aligned}\widehat{W[\lambda]f}(\omega) = \widehat{f * \psi}(\omega) &= \left(\left(\widehat{f}(\omega)e^{i2^j\omega}\right)^* + \widehat{f}(\omega)e^{i2^j\omega}\right)\chi_{[2^{-(j+1)},2^{-j}]}(\omega)\\ &= 2\,\mathrm{Re}\,\widehat{f}(\omega)e^{i2^j\omega}\chi_{[2^{-(j+1)},2^{-j}]}(\omega) = 2\,\mathrm{Re}\,\widehat{T_{2^j}f}(\omega)\chi_{[2^{-(j+1)},2^{-j}]}(\omega)\end{aligned}$$

where $T_{2^j}f(x) = f(x-2^j)$. Thus $\widehat{W[\lambda]f}(\omega)$ is supported in the interval $[2^{-(j+1)}, 2^{-j}]$. When applying the modulus operator, high frequencies will be mapped to lower frequencies. To see this, note that

$$(U[\lambda]f)^2 = |W[\lambda]f|^2 = W[\lambda]f\,(W[\lambda]f)^*.$$

Taking the Fourier transform yields

$$\begin{aligned}\widehat{(U[\lambda]f)^2}(\omega) &= \frac{1}{2\pi}\int W[\lambda]f(x)\,(W[\lambda]f(x))^*\,e^{-i\omega x}dx\\ &= \frac{1}{2\pi}\int W[\lambda]f(x)\left(W[\lambda]f(x)e^{i\omega x}\right)^* dx.\end{aligned}$$

The Plancherel equality(A.10) implies that

$$\begin{aligned}\widehat{(U[\lambda]f)}^2(\omega) &= \frac{1}{2\pi}\int \widehat{W[\lambda]f}(\xi)(\widehat{W[\lambda]fe^{i\omega x}})^*(\xi)d\xi \\ &= \frac{1}{2\pi}\int \widehat{W[\lambda]f}(\xi)\widehat{W[\lambda]f}^*(\xi-\omega)d\xi.\end{aligned}$$

As $\widehat{W[\lambda]f}(\xi)$ is non-zero for $\xi \in [2^{-(j+1)}, 2^{-j}]$, the product $\widehat{W[\lambda]f}(\xi)\widehat{W[\lambda]f}^*(\xi-\omega)$ is non-zero only when $\omega \in [-2^{-(j+1)}, 2^{-(j+1)}]$. Hence the modulus of the wavelet transform has frequencies located in the interval $[-2^{-(j+1)}, 2^{-(j+1)}]$.

Let $\lambda' = 2^k$. We want to show that $\|W[\lambda']U[\lambda]f\|_{L^2}$ is non-zero only when $|\lambda'|^{-1} < |\lambda|^{-1} \iff 2^{-k} < 2^{-j}$. By Parsevals equality(A.11) we have

$$\begin{aligned}\|W[\lambda']U[\lambda]f\|_{L^2}^2 &= \int ||f*\psi_\lambda|\psi_{\lambda'}(x)|^2dx \\ &= \int |\widehat{U[\lambda]f}(\omega)\widehat{\psi_{\lambda'}}(\omega)|^2d\omega.\end{aligned}$$

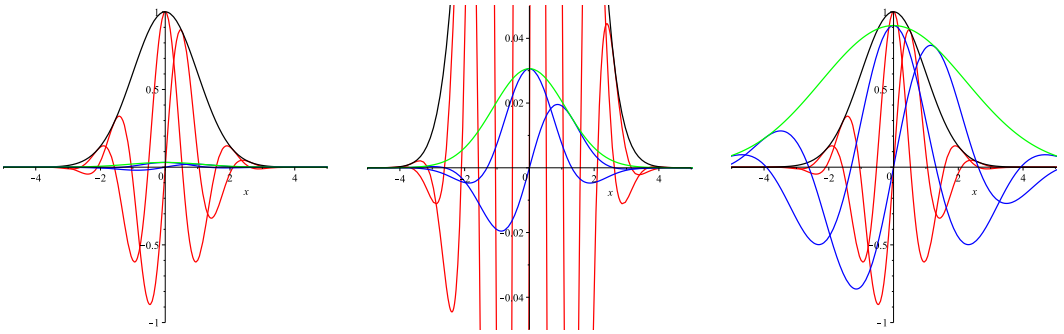
Since $\operatorname{supp}\widehat{(U[\lambda]f)}^2 \subseteq [-2^{-(j+1)}, 2^{-(j+1)}]$ and $\operatorname{supp}\widehat{\psi_{\lambda'}}(\omega) \subseteq [-2^{-k}, -2^{-(k+1)}] \cup [2^{-(k+1)}, 2^{-k}]$, the two functions have overlapping support only if $2^{-(k+1)} < 2^{-(j+1)} \iff 2^{-k} < 2^{-j}$. □

If ψ is the Morlet wavelet given in (3.35), then ψ is not compactly supported. However, the Gaussian envelope ensures that the energy is concentrated around the frequency ξ. A similar argument as in the proof above, shows that most of the energy is concentrated along frequency-decreasing paths. In Figure 4.8 this is illustrated with the signal $f(x) = \delta(x)$. The red graphs show the real and imaginary parts of $f*\psi_\lambda$, with $|\lambda| = 2^0$. The black graph is the corresponding envelope obtained by applying the modulus operation. In Figure 4.8(a) and 4.8(b), the blue graphs are the real and imaginary parts of $|f*\psi_\lambda|*\psi_{\lambda'}$, with $|\lambda'| = 2^{-1}$. The green graph is the corresponding envelope. In Figure 4.8(c) the same situation is illustrated but now with $\lambda' = 2^1$, i.e along a frequency-decreasing path. It is easy to see that the energy contained in the coefficients along the path (λ, λ') is much smaller when $|\lambda'| = 2$ compared to the energy along the path (λ, λ') when $|\lambda'| = 1/2$.

When we compute the windowed scattering transform numerically, we will take advantage of this observation. In other words, we will compute scattering coefficients only along frequency-decreasing paths. However in this chapter where we cover the theoretical part, we will include all paths if not otherwise specified.

4.3 The scattering metric

The goal is to construct an operator $\Phi : L^2(\mathbb{R}^2) \to \mathcal{H}$ which reduces intraclass variability, so that the induced metric on the Hilbert space $\mathcal{H}$ can be used to measure distances between images. The distance between images belonging to the same class should be small compared to the distance between images belonging to different classes.



(a) Illustration of the size of the coefficients along a frequency-increasing path.

(b) Zooming in on Figure 4.8(a).

(c) Illustration of the size of the coefficients along a frequency-decreasing path.

FIGURE 4.8: Illustration of the size of the coefficients along frequency-increasing and frequency-decreasing paths.

So far, we have defined the windowed scattering transform of signals belonging to $L^2(\mathbb{R}^2)$. When comparing two signals f and g, we compute the windowed scattering transform along every path $p \in \mathcal{P}_J$. The output of the windowed scattering transform belongs to the product space generated by copies of $L^2(\mathbb{R}^2)$, with norm defined in (4.16).

Let $f, g \in L^2(\mathbb{R}^2)$ be two images and define the scattering metric as

$$d_J(f,g) = \|S_J[\mathcal{P}_J]f - S_J[\mathcal{P}_J]g\|. \tag{4.17}$$

For admissible wavelets, the scattering metric possesses the following properties:

- Stability to additive noise. If $f(x) = f_0(x) + n(x)$, where n represents the noise in the image, then

$$d_J(f, f_0) = \|S_J[\mathcal{P}_J]f - S_J[\mathcal{P}_J]f_0\| \leq \|f - f_0\|_{L^2} = \|n\|_{L^2}.$$

- Locally translation invariance. As $J \to \infty$ the scattering metric converges to a translation invariant metric,

$$\lim_{J\to\infty} d_J(f, T_c f) = 0. \tag{4.18}$$

- Stability with respect to the action of C^2 diffeomorphisms. Let $\tau \in C^2(\mathbb{R}^2)$, and consider the set of paths in $\mathcal{P}_J$ of length strictly smaller then m, then

$$d_J(f, T_\tau f) \leq Cm\|f\|_{L^2} \left(\sup_x |\nabla\tau(x)| + \sup_x |H\tau(x)| \right), \tag{4.19}$$

  for all $f \in L^2(\mathbb{R}^2)$ with compact support.

In the following we will assume that the wavelet transform W_J satisfies the condition given in Proposition 3.13 with $\epsilon = 0$, that is $\|W_J f\| = \|f\|_{L^2}$.

We will start to show stability to additive noise. The first lemma shows that the one-step propagator is non-expansive.

Lemma 4.6. *The one-step propagator U_J is non-expansive and preserves the norm.*

Proof. Let $f, g \in L^2(\mathbb{R}^2)$. The modulus operator is non-expansive in the sense that for any $a, b \in \mathbb{C}$ we have $||a| - |b|| \leq |a - b|$. Consequently,

$$\begin{aligned}
\|U_J f - U_J g\|^2 &= \|A_J f - A_J g\|^2 + \sum_{\lambda \in \Lambda_J} \|U[\lambda] f - U[\lambda] g\|^2 \\
&= \|A_J f - A_J g\|^2 + \sum_{\lambda \in \Lambda_J} \| |W[\lambda] f| - |W[\lambda] g| \|^2 \\
&\leq \|A_J f - A_J g\|^2 + \sum_{\lambda \in \Lambda_J} \|W[\lambda] f - W[\lambda] g\|^2 \\
&= \|W_J f - W_J g\|^2 = \|W_J (f - g)\|^2 = \|f - g\|_{L^2}^2.
\end{aligned}$$

The last equality follows from the fact that W_J is assumed to be unitary. If $g = 0$ we see that $\|U_J f\| = \|f\|_{L^2}$, hence U_J is preserves the norm. □

Since the windowed scattering transform is obtained by iteratively applying U_J it follows that S_J is also non-expansive.

Theorem 4.7. *Let $J \in \mathbb{Z}$, then for all $f, g \in L^2(\mathbb{R}^2)$*

$$\|S_J[\mathcal{P}_J] f - S_J[\mathcal{P}_J] g\| \leq \|f - g\|_{L^2}, \tag{4.20}$$

that is, the windowed scattering transform is non-expansive.

Proof. By Lemma 4.6, $\|f - g\|_{L^2}^2 \geq \|U_J f - U_J g\|^2$. Since $U_J f = \{A_J f, U[\Lambda_J^1] f\} = \{S_J[\Lambda_J^0] f, U[\Lambda_J^1] f\}$,

$$\begin{aligned}
\|f - g\|_{L^2}^2 &\geq \|U_J f - U_J g\|^2 \\
&= \|S_J[\Lambda_J^0] f - S_J[\Lambda_J^0] g\|^2 + \|U[\Lambda_J^1] f - U[\Lambda_J^1] g\|^2 \\
&\geq \|S_J[\Lambda_J^0] f - S_J[\Lambda_J^0] g\|^2 + \|U_J U[\Lambda_J^1] f - U_J U[\Lambda_J^1] g\|^2 \\
&= \|S_J[\Lambda_J^0] f - S_J[\Lambda_J^0] g\|^2 + \|S_J[\Lambda_J^1] f - S_J[\Lambda_J^1] g\|^2 + \|U[\Lambda_J^2] f - U[\Lambda_J^2] g\|^2 \\
&\geq \cdots \geq \sum_{k=0}^{m-1} \|S_J[\Lambda_J^k] f - S_J[\Lambda_J^k] g\|^2 + \|U[\Lambda_J^m] f - U[\Lambda_J^m] g\|^2 \\
&\geq \sum_{k=0}^{m-1} \|S_J[\Lambda_J^k] f - S_J[\Lambda_J^k] g\|^2 + \|U_J U[\Lambda_J^m] f - U_J U[\Lambda_J^m] g\|^2 \\
&= \sum_{k=0}^{m} \|S_J[\Lambda_J^k] f - S_J[\Lambda_J^k] g\|^2 + \|U[\Lambda_J^{m+1}] f - U[\Lambda_J^{m+1}] g\|^2.
\end{aligned}$$

The inequality

$$\begin{aligned}
\sum_{k=0}^{m} \|S_J[\Lambda_J^k] f - S_J[\Lambda_J^k] g\|^2 &\leq \sum_{k=0}^{m} \|S_J[\Lambda_J^k] f - S_J[\Lambda_J^k] g\|^2 + \|U[\Lambda_J^{m+1}] f - U[\Lambda_J^{m+1}] g\|^2 \\
&\leq \|f - g\|_{L^2}^2
\end{aligned}$$

is true for any $m \in \mathbb{N}$. Letting $m \to \infty$, we see that

$$\|S_J[\mathcal{P}_J]f - S_J[\mathcal{P}_J]g\|^2 = \sum_{k=0}^{\infty} \|S_J[\Lambda_J^k]f - S_J[\Lambda_J^k]g\|^2 \leq \|f-g\|_{L^2}^2.$$

□

Theorem 4.7 implies that the windowed scattering transform is Lipschitz continuous, and thus stable to additive noise. In fact, one can prove that if W_J is unitary, then the windowed scattering transform also preserves the energy in the signal, $\|S_J[\mathcal{P}_J]f\| = \|f\|_{L^2}$. Since U_J preserves the norm, similar calculations as in the proof in Theorem 4.7 show that for any $m \in \mathbb{N}$,

$$\|f\|_{L^2}^2 = \sum_{k=0}^{m} \|S_J[\Lambda_J^k]f\|^2 + \|U[\Lambda_J^{m+1}]f\|^2.$$

If $m \to \infty$, then

$$\|f\|_{L^2}^2 = \|S_J[\mathcal{P}_J]f\|^2 + \lim_{m\to\infty} \|U[\Lambda_J^{m+1}]f\|^2.$$

Hence S_J preserves the energy if the energy vanishes along increasing paths, that is $\lim_{m\to\infty} \|U[\Lambda_J^{m+1}]f\|^2 = 0$. To show that $\lim_{m\to\infty} \|U[\Lambda_J^{m+1}]f\|^2 = 0$ we need the following lemma:

Lemma 4.8. *Suppose $f \in L^2(\mathbb{R}^2)$. If ψ is an admissible scattering wavelet and*

$$\|f\|_w^2 = \sum_{j=0}^{\infty} \sum_{r\in G^+} j\|W[2^{-j}r]f\|^2 < \infty, \tag{4.21}$$

then

$$\frac{\alpha}{2}\|U[\mathcal{P}_J]f\|^2 \leq \max(J+1,1)\|f\|_{L^2}^2 + \|f\|_w^2, \tag{4.22}$$

where α is given by (3.47). Moreover, there exists a sequence f_n, with $\|f_n\|_w < \infty$ and $\lim_{n\to\infty} \|f - f_n\|_{L^2} = 0$.

Proof. We will prove the last part of this lemma, i.e that the space of functions for which $\|f\|_w < \infty$ is a dense subspace in $L^2(\mathbb{R}^2)$. Proof of the first part can found in [Mal12].

If ϕ is the scaling function from the averaging operator, let $\phi_{2^{-n}}(x) = 2^{2n}\phi(2^n x)$ and define

$$f_n(x) = f * \phi_{2^{-n}}(x).$$

Since $\phi \in L^1(\mathbb{R}^2)$, $\widehat{\phi}(0) = 1$ and $\phi \geq 0$, the sequence $\phi_{2^{-n}}$ is an approximate identity, and hence we have $\lim_{n\to\infty} \|f - f_n\|_{L^2} = 0$. To show that $\|f_n\|_w < \infty$, note that from Remark 3.12 we have $|\psi(\omega)| = \mathcal{O}(|\omega|)$ and $|\omega||\phi(\omega)| < \infty$. Let C

be a generic constant that may change value from one line to the next, then

$$\begin{aligned}
\|W[2^{-j}r]f_n\|^2 &= \int_{\mathbb{R}^2} |f_n * \psi_{2^{-j}r}(x)|^2 dx \\
&= \int_{\mathbb{R}^2} |\widehat{f * \phi_{2^{-n}}}(\omega)|^2 |\widehat{\psi_{2^{-j}r}}(\omega)|^2 d\omega \\
&= \int_{\mathbb{R}^2} |\widehat{f}(\omega)|^2 |\widehat{\phi}(2^{-n}\omega)|^2 |\widehat{\psi}(2^{-j}r^{-1}\omega)|^2 d\omega \\
&\leq C \int_{\mathbb{R}^2} |\widehat{f}(\omega)|^2 |\widehat{\phi}(2^{-n}\omega)|^2 |2^{-2j}\omega|^2 d\omega \\
&= C 2^{2n-2j} \int_{\mathbb{R}^2} |\widehat{f}(\omega)|^2 |\widehat{\phi}(2^{-n}\omega)|^2 |2^{-n}\omega|^2 d\omega \\
&\leq C 2^{2n-2j} \int_{\mathbb{R}^2} |\widehat{f}(\omega)|^2 d\omega = C 2^{2n-2j}.
\end{aligned}$$

The series $\sum_{j=0}^{\infty} j 2^{-2j}$ is convergent, hence $\|f_n\|_w < \infty$ by comparison test.

□

Theorem 4.9. *If ψ is an admissible scattering wavelet, and W_J satisfies (3.13) with $\epsilon = 0$, then for any $f \in L^2(\mathbb{R}^2)$,*

$$\lim_{m\to\infty} \|U[\Lambda_J^m]f\|^2 = \lim_{m\to\infty} \sum_{n=m}^{\infty} \|S_J[\Lambda_J^n]f\|^2 = 0, \tag{4.23}$$

and

$$\|S_J[\mathcal{P}_J]f\| = \|f\|_{L^2}. \tag{4.24}$$

Proof. We will show that $\lim_{m\to\infty} \|U[\Lambda_J^m]f\|^2 = 0$. If $\|f\|_w < \infty$ then by Lemma 4.8

$$\|U[\mathcal{P}_J]f\|^2 = \sum_{m=0}^{\infty} \|U[\Lambda_J^m]f\|^2 \leq \frac{2}{\alpha} \max(J+1, 1)\|f\|_{L^2}^2 + \|f\|_w^2 < \infty,$$

and hence $\lim_{m\to\infty} \|U[\Lambda_J^m]f\|^2 = 0$.

If $\|f\|_w = \infty$, we can find a sequence f_n, with $\|f_n\|_w < \infty$ and $\lim_{n\to\infty} \|f - f_n\|_{L^2} = 0$. By the triangle inequality

$$\|U[\Lambda_J^m]f\| \leq \|U[\Lambda_J^m]f - U[\Lambda_J^m]f_n\| + \|U[\Lambda_J^m]f_n\|,$$

and since $U[\Lambda_J^m]$ is non-expansive, we have

$$\|U[\Lambda_J^m]f\| \leq \|U[\Lambda_J^m]f - U[\Lambda_J^m]f_n\| + \|U[\Lambda_J^m]f_n\| \leq \|f - f_n\|_{L^2} + \|U[\Lambda_J^m]f_n\|.$$

Let $\varepsilon > 0$ be given. Then we can find $N \in \mathbb{N}$ so that $\|f - f_n\|_{L^2} < \varepsilon/2$ whenever $n \geq N$. By the first part we can find $M \in \mathbb{N}$ so that if $m \geq M$ we have $\|U[\Lambda_J^m]f_N\| < \varepsilon/2$. Consequently, if $m \geq M$ we have

$$\|U[\Lambda_J^m]f\| \leq \|f - f_N\| + \|U[\Lambda_J^m]f_N\| < \varepsilon/2 + \varepsilon/2 = \varepsilon,$$

hence $\lim_{m\to\infty} \|U[\Lambda_J^m]f\| = 0$. □

In practical applications to image processing, numerical evidence shows (see Table 5.1) that the convergence of $\|U[\Lambda_J^m]f\|$ is exponential as a function of m. Therefore, limiting computations to only a finite number of layers will give a small loss of information.

We will now see that the scattering metric is a locally translation invariant metric (4.3), and converges to a fully translation invariant metric as $J \to \infty$. The next lemma implies that the scattering distance between two signals is decreasing as J is increasing.

Lemma 4.10. *For all $f, g \in L^2(\mathbb{R}^2)$ and any $J \in \mathbb{Z}$,*

$$\|S_{J+1}[\mathcal{P}_{J+1}]f - S_{J+1}[\mathcal{P}_{J+1}]g\| \leq \|S_J[\mathcal{P}_J]f - S_J[\mathcal{P}_J]g\|. \tag{4.25}$$

Proof. See [Mal12] □

This result shows that the sequence $\|S_J[\mathcal{P}_J]f - S_J[\mathcal{P}_J]g\|$ is monotonically decreasing, and hence the convergence is guaranteed by the monotone convergence theorem, whenever $\|S_J[\mathcal{P}_J]f - S_J[\mathcal{P}_J]g\| \leq \|f - g\|_{L^2} < \infty$.

The scattering operator S_J commutes with the translation operator, $S_J[\mathcal{P}_J]T_c = T_c S_J[\mathcal{P}_J]$. To see this note that

$$\begin{aligned} S_J[\mathcal{P}_J]T_c f(x) = A_J U[\mathcal{P}_J]T_c f(x) &= \int U[\mathcal{P}_J]f(u-c)\phi_J(x-u)du \\ &= \int U[\mathcal{P}_J]f(u)\phi_J(x-c-u)du = S_J[\mathcal{P}_J]f(x-c) \\ &= T_c S_J[\mathcal{P}_J]f(x). \end{aligned}$$

Therefore,

$$\begin{aligned} \|S_J[\mathcal{P}_J]T_c f - S_J[\mathcal{P}_J]f\| &= \|T_c S_J[\mathcal{P}_J]f - S_J[\mathcal{P}_J]f\| \\ &= \|T_c A_J U[\mathcal{P}_J]f - A_J U[\mathcal{P}_J]f\| \\ &\leq \|T_c A_J - A_J\|_{op} \|U[\mathcal{P}_J]f\|. \end{aligned}$$

The operator norm $\|T_c A_J - A_J\|_{op}$ is given by

$$\|T_c A_J - A_J\|_{op} = \sup\left\{\|(T_c A_J - A_J)U[\mathcal{P}_J]f\| : \|U[\mathcal{P}_J]f\| = 1\right\}.$$

Moreover, we have the following result:

Lemma 4.11. *Suppose $\tau \in C^2(\mathbb{R}^2)$, with $\sup_x |\nabla\tau(x)| < 1/2$. Then there exists a constant $C > 0$ such that*

$$\|T_\tau A_J - A_J\|_{op} \leq C 2^{-J} \sup_x |\tau(x)|. \tag{4.26}$$

Proof. First note that

$$T_\tau A_J f(x) - A_J f(x) = \int f(y)\left(\phi_{2^J}(x - \tau(x) - y) - \phi_{2^J}(x-y)\right) dy.$$

The result follows by applying Schur test (A.1) to the kernel

$$K(x,y) = \phi_{2^J}(x - \tau(x) - y) - \phi_{2^J}(x - y),$$

that is, if

$$\sup_{x\in\mathbb{R}} \int |K(x,y)|dy \le K_1, \quad \text{and} \quad \sup_{y\in\mathbb{R}} \int |K(x,y)|dx \le K_2, \tag{4.27}$$

then $T_\tau A_J - A_J$ satisfies

$$\|T_\tau A_J - A_J\|_{op} \le \sqrt{K_1 K_2}. \tag{4.28}$$

By Taylor expansion we see that

$$|K(x,y)| \le \sup_x |\tau(x)| \int_0^1 |\nabla\phi_{2^J}(x - y - t\tau(x))|dt.$$

Since $\nabla\phi_{2^J}(x) = 2^{-2J-J}\nabla\phi(2^{-J}x)$,

$$\begin{aligned}
\int |K(x,y)|dy &\le \sup_x |\tau(x)| \int\int_0^1 |\nabla\phi_{2^J}(x - y - t\tau(x))|dtdy \\
&= \sup_x |\tau(x)| \int_0^1 \int |\nabla\phi_{2^J}(x - y - t\tau(x))|dydt \\
&= \sup_x |\tau(x)| 2^{-2J-J} \int |\nabla\phi(2^{-J}u)|du \\
&= \sup_x |\tau(x)| 2^{-J} \|\nabla\phi\|_{L^1} = K_1.
\end{aligned}$$

The interchanging of integrals is justified by Tonelli's theorem (A.2). Note that by the assumptions in Remark 3.12 we have $\nabla\phi \in L^1$. Similarly,

$$\int |K(x,y)|dx \le \sup_x |\tau(x)| \int\int_0^1 |\nabla\phi_{2^J}(x - y - t\tau(x))|dtdx.$$

The change of variables $u = x - y - t\tau(x)$, gives the Jacobian matrix $D = I - t\nabla\tau(x)$, where I denotes the identity matrix. By assumption $\sup|\nabla\tau| < 1/2$, which implies $\det D \ge (1 - \sup|\nabla\tau|)^2 \ge 2^{-2}$. Hence

$$\begin{aligned}
\int |K(x,y)|dx &\le (\det D)^{-1} \sup_x |\tau(x)| \int\int_0^1 |\nabla\phi_{2^J}(x - y - t\tau(x))|dtdx \\
&= \sup_x |\tau(x)| 2^{-J} \|\nabla\phi\|_{L^1} 2^2 = K_2.
\end{aligned}$$

The Schur test implies that

$$\begin{aligned}
\|T_\tau A_J - A_J\|_{op} \le \sqrt{K_1 K_2} &= \sqrt{\sup_x |\tau(x)|^2 2^{-2J+2} \|\nabla\phi\|_1^2} \\
&= \sup_x |\tau(x)| 2^{-J+1} \|\nabla\phi\|_{L^1} = C 2^{-J} \sup_x |\tau(x)|.
\end{aligned}$$

□

With $\tau(x) = c$ we see that

$$\|S_J[\mathcal{P}_J]T_c f - S_J[\mathcal{P}_J]f\| \leq C|c|2^{-J}\|U[\mathcal{P}_J]f\|.$$

Theorem 4.12. *If ψ is an admissible wavelet, then for any $f \in L^2(\mathbb{R}^2)$*

$$\lim_{J\to\infty} \|S_J[\mathcal{P}_J]T_c f - S_J[\mathcal{P}_J]f\| = 0.$$

Proof. If $J > 1$, then by Lemma 4.8,

$$\|U[\mathcal{P}_J]\|^2 \leq \frac{2}{\alpha}\left((J+1)\|f\|_{L^2}^2 + \|f\|_w^2\right).$$

If $\|f\|_w < \infty$ then

$$\lim_{J\to\infty} \|S_J[\mathcal{P}_J]T_c f - S_J[\mathcal{P}_J]f\| \leq \lim_{J\to\infty} C^2|c|^2 2^{-2J}\frac{2}{\alpha}\left((J+1)\|f\|_{L^2}^2 + \|f\|_w^2\right) = 0.$$

If $\|f\|_w = \infty$, there exists a sequence (f_n) with $\|f_n\|_w < \infty$ such that $\lim_{n\to\infty}\|f - f_n\|_{L^2} = 0$. Hence

$$\begin{aligned}
\|T_c S_J[\mathcal{P}_J]f - S_J[\mathcal{P}_J]f\| &\leq \|T_c S_J[\mathcal{P}_J]f - T_c S_J[\mathcal{P}_J]f_n\| + \|S_J[\mathcal{P}_J]f_n - S_J[\mathcal{P}_J]f\| \\
&\quad + \|T_c S_J[\mathcal{P}_J]f_n - S_J[\mathcal{P}_J]f_n\| \\
&\leq (\|T_c\|_{op} + 1)\,\|S_J[\mathcal{P}_J]f - S_J[\mathcal{P}_J]f_n\| \\
&\quad + \|T_c S_J[\mathcal{P}_J]f_n - S_J[\mathcal{P}_J]f_n\| \\
&= 2\|S_J[\mathcal{P}_J]f - S_J[\mathcal{P}_J]f_n\| + \|T_c S_J[\mathcal{P}_J]f_n - S_J[\mathcal{P}_J]f_n\| \\
&\leq 2\|f - f_n\|_{L^2} + \|T_c S_J[\mathcal{P}_J]f_n - S_J[\mathcal{P}_J]f_n\|.
\end{aligned}$$

Let $\varepsilon > 0$ be given. Then there exists $N \in \mathbb{N}$ so that for $n \geq N$ we have $\|f - f_n\|_{L^2} < \varepsilon/4$. Since $\|f_N\|_w < \infty$, the first part implies that there exists $\tilde{J}$ so that for $J \geq \tilde{J}$ we have $\|T_c S_J[\mathcal{P}_J]f_N - S_J[\mathcal{P}_J]f_N\| < \varepsilon/2$. Hence for $J \geq \tilde{J}$,

$$\begin{aligned}
\|T_c S_J[\mathcal{P}_J]f - S_J[\mathcal{P}_J]f\| &\leq 2\|f - f_N\|_{L^2} + \|T_c S_J[\mathcal{P}_J]f_N - S_J[\mathcal{P}_J]f_N\| \\
&< 2\frac{\epsilon}{4} + \frac{\epsilon}{2} = \varepsilon.
\end{aligned}$$

Thus

$$\lim_{J\to\infty} \|T_c S_J[\mathcal{P}_J]f - S_J[\mathcal{P}_J]f\| = 0.$$

□

This shows that as $J \to \infty$, the scattering metric converges to a translation invariant metric.

Remark 4.13. In practice, signals are compactly supported, and therefore one need not choose J very large to obtain a translation invariant metric. If f is an image with $N \times N$ pixels, then choosing J so that $N < 2^J$ will give a translation invariant metric. However, for classification purposes we see, in view of Lemma 4.10, that choosing J too large will shorten the distance between all signals, which

may cause misclassification. The scale J should therefore be chosen with respect to the largest translation in the database.

We have now shown that the scattering metric is stable to additive noise and converges to a translation invariant operator. It remains to show that it is also stable to the action of diffeomorphisms. As we are interested in application to images which are real and compactly supported signals, we only consider this special case. We will state the result and refer to [Mal12] for details of the proof.

Theorem 4.14. *Let*

$$\|U[\mathcal{P}_J]f\|_1 = \sum_{m=0}^{\infty} \|U[\Lambda_J^m]f\|, \tag{4.29}$$

and denote $\mathcal{P}_J^{m-1}$ the subset of $\mathcal{P}_J$ of paths of length strictly smaller than m. Then for any compact set $\Omega \subseteq \mathbb{R}^2$ there exists $C > 0$, such that for all $f \in L^2(\mathbb{R}^2)$ supported in Ω with $\|U[\mathcal{P}_J]f\|_1 < \infty$, and for all $\tau \in C^2(\mathbb{R}^2)$ with $\sup_{x\in\Omega} |\nabla\tau(x)| \leq 1/2$,

$$\|S_J[\mathcal{P}_J^{m-1}]T_\tau f - S_J[\mathcal{P}_J^{m-1}]f\| \leq Cm\|f\|_{L^2}\left(2^{-J}\sup_{x\in\Omega}|\tau(x)| + \sup_{x\in\Omega}|\nabla\tau(x)| + \sup_{x\in\Omega}|H\tau(x)|\right). \tag{4.30}$$

The term $2^{-J}\sup_{x\in\Omega}|\tau(x)|$ corresponds to the local translation invariance. If J is chosen so that $2^{-J}\sup_{x\in\Omega}|\tau(x)| \leq \sup_{x\in\Omega}|\nabla\tau(x)| + \sup_{x\in\Omega}|H\tau(x)|$, then for a possible different constant C, we have

$$\|S_J[\mathcal{P}_J^{m-1}]T_\tau f - S_J[\mathcal{P}_J^{m-1}]f\| \leq Cm\|f\|_{L^2}\left(\sup_{x\in\Omega}|\nabla\tau(x)| + \sup_{x\in\Omega}|H\tau(x)|\right). \tag{4.31}$$

For application to image processing, numerical evidence shows that the scattering energy becomes negligible over paths of length larger than $m = 3$, see e.g Table 5.1 in Chapter 5. Therefore, for all practical purposes we can apply (4.31) with $m = 4$.

The scattering metric will reduce variability within each class, if the variability is primarily due to translation, additive noise, and deformations. In the next chapter we will see how to compute the windowed scattering transform numerically. Eventually we will see how we can classify images based on their scattering coefficients and the area of inflammation.

Chapter 5

Application to digital images

At this point, we have build up the theory behind the windowed scattering transform and edge-detection. In order to apply this theory to digital images, we need a discrete formulation.

The first section gives a short introduction to digital image processing, and we will see how images can be represented digitally. Thereafter, a review of how to compute the area of inflammation and the scattering coefficients numerically will be conducted. In the last section we will present the classification algorithms.

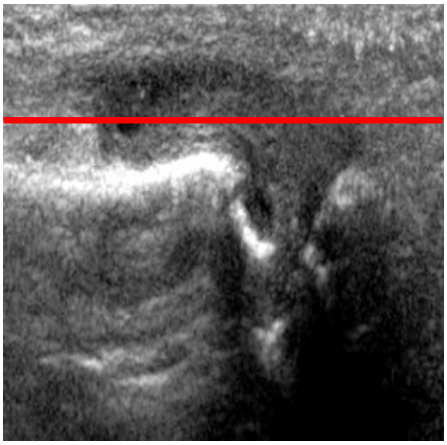
5.1 Digital image processing

A grayscale image f is a continuous function with finite energy, $f : \Omega \to \mathbb{R}$ where $\Omega \subset \mathbb{R}^2$ is compact. The value $f(x, y)$ is called the intensity at the point (x, y). To represent, and store an image digitally on a computer, we need to convert the continuous signal into a digital signal. This involves two processes called sampling and quantization [GW08, p.74]. Sampling is the process of digitalizing the coordinate values, while quantization is about digitalizing the intensity.

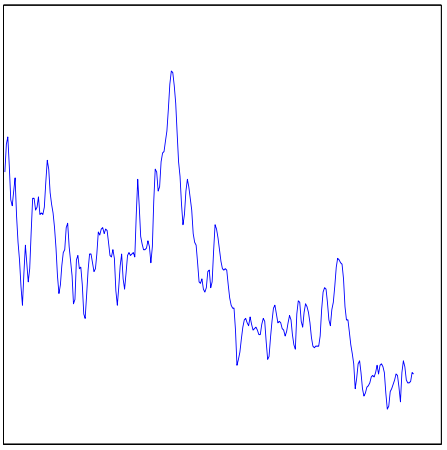
In Figure 5.1, the process of digitalizing an image is illustrated. The image function is sampled along every row and column at evenly spaced coordinates. Figure 5.1(a) shows a continuous image, and in Figure 5.1(d) we see the resulting digital image. The digital image is arranged in a $N \times K$ matrix where each element in this matrix is called a pixel. Each pixel is assigned a value corresponding to the quantized intensity. This means that we can view the digitalizing of the image f as a mapping

$$L^2(\mathbb{R}^2) \xrightarrow[\text{digitalizing}]{} \mathrm{M}_{N,K}(\mathbb{R}),$$

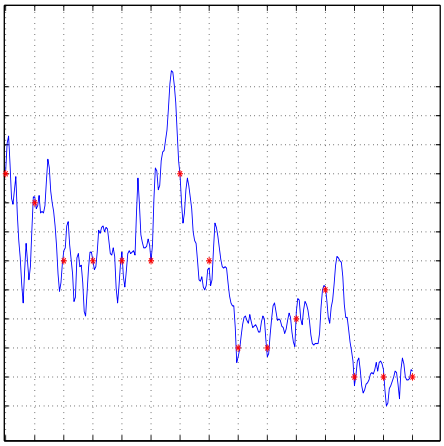
where $\mathrm{M}_{N,K}(\mathbb{R})$ denotes the set of all $N \times K$ matrices over $\mathbb{R}$. In fact, due to the quantization, the range of the digitalized image f is a finite subset of $\mathbb{R}$, but this is not important for us.



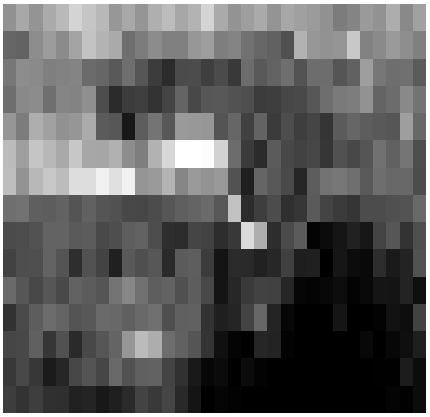
(a) Continuous image.



(b) A scan of the red line in Figure 5.1(a).



(c) Sampling and quantization of the signal in Figure 5.1(b).



(d) Digitalized image, sampled along every row and colum.

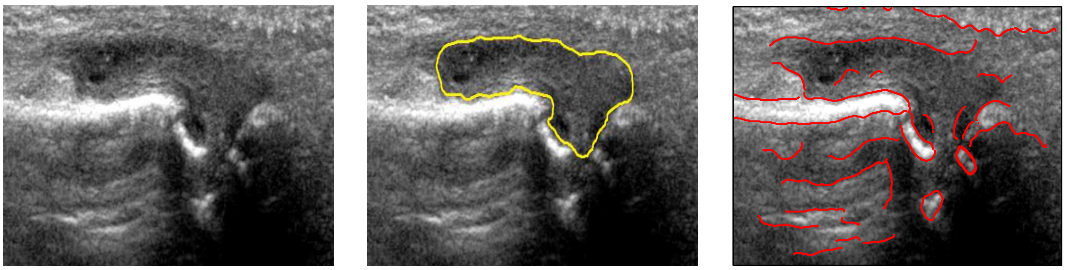
FIGURE 5.1: Digitalizing a continuous image.

5.2 Area of inflammation

In this section we will briefly review how to calculate the area of inflammation numerically.

5.2.1 Problems with finding the correct edges

To calculate the area, we will apply the Canny edge-detector to find the boundary of the inflamed region, and use Greens formula to estimate the area. As this method is highly dependent on finding the correct edges, we would like to address some issues relating to detection of the correct edges. Thereafter, we will present the solution used in this thesis.



(a) Original image. (b) Annotated image. (c) Edge-detector applied to image in Figure 5.2(a).

FIGURE 5.2: Example of an annotated image, and illustration of edges found by the Canny edge-detector.

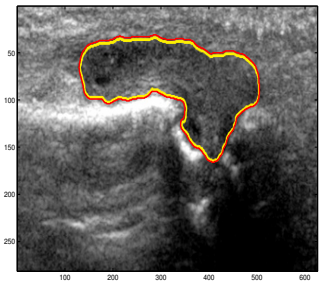


FIGURE 5.3: Canny edge-detector applied to an annotated image. Parameters used are $T_{\text{low}} = 0.09$, $T_{\text{high}} = 0.68$, and $s = 1.8$.

Consider the US image shown in Figure 5.2(a). The correct boundary of the synovitis is shown in the annotated image in Figure 5.2(b). By testing the edge-detector for different values of $T_{\text{high}}, T_{\text{low}}$ and scale s, we find that the edges highlighted in Figure 5.2(c) give an indication of where the boundary of the synovitis is located. The problem is however, that the curve defining the boundary is not connected. Moreover, additional edges make it difficult for an algorithm to determine which edges that should be selected as the boundary-edges. The reason for this problem is because there are other objects, such as bones, tendons and fat present in the image. In addition, noise makes the transition between the different objects blurry, so that boundary-lines are broken.

As we are mainly interested in investigating whether the area defines an efficient classifier we will apply the edge-detection algorithm to annotated images. An example of the edge-detector applied to an annotated image is shown in Figure 5.3. Here the yellow curve is the annotated curve, while the red curve is the output from the edge detector. The parameters used in this case is $T_{\text{high}} = 0.68, T_{\text{low}} = 0.09$ and $s = 1.8$. In this way we are able to detect the annotated boundary, and thus find the correct area.

5.2.2 Calculating the area

We will calculate the area using Greens formula,

$$\text{Area} = \int_D dA = \frac{1}{2}\int_C xdy - ydx. \tag{5.1}$$

The first step is to apply the Canny edge-detector. A built-in function in Matlab called edge() is used in this thesis. This will, for appropriate values for the thresholds $T_{\text{high}}, T_{\text{low}}$ and scale s, detect the annotated boundary. The output will be a sequence of N points (x_i, y_i), which approximates the enclosing boundary curve C in (5.1) as a piecewise linear, simple, closed curve. To ensure that the curve is closed we set $(x_1, y_1) = (x_N, y_N)$. The area can then be computed by the following formula

$$\text{Area} = \frac{1}{2}\int_C xdy - ydx = \frac{1}{2}\sum_{i=1}^{N-1}\int_{(x_i,y_i)}^{(x_{i+1},y_{i+1})} xdy - ydx \tag{5.2}$$

$$= \frac{1}{2}\sum_{i=1}^{N-1}\left[\frac{x_i + x_{i+1}}{2}(y_{i+1} - y_i) - \frac{y_i + y_{i+1}}{2}(x_{i+1} - x_i)\right]. \tag{5.3}$$

Here the integral $\int_{y_i}^{y_{i+1}} xdy$ is approximated by $\frac{x_i+x_{i+1}}{2}(y_{i+1} - y_i)$, which is also known as the trapezoidal rule. Since the curve C is piecewise linear, the approximation is exact. Hence, any error in calculating the area is due to the annotation and the edge-detector.

5.3 The windowed scattering transform

In this section we will see how to compute the scattering coefficients of a digital image. The software used is called ScatNet and can be downloaded from [ASM+14].

5.3.1 Filter bank

We will use the Morlet wavelet, with a Gaussian low-pass filter ϕ. For $x \in \mathbb{R}^2$, the low pass filter is given by

$$\phi(x) = \frac{1}{2\pi\sigma_\phi^2} e^{-\frac{|x|^2}{2\sigma_\phi^2}}. \tag{5.4}$$

The Morlet wavelet, rotated by an angle θ is implemented in the following way

$$\psi_\theta(x) = \exp\left\{-\frac{x^T r_{-\theta}\begin{pmatrix}1 & 0\\ 0 & s^2\end{pmatrix} r_\theta x}{2\sigma_\psi^2}\right\}\left[\exp\left(i\langle\begin{pmatrix}\xi & 0\end{pmatrix}, r_\theta x\rangle\right) - \exp\left(-\frac{\sigma_\psi^2\xi^2}{2}\right)\right]. \tag{5.5}$$

Note that the wavelet is not normalized. Here r_θ is a rotation matrix

$$r_\theta = \begin{pmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{pmatrix}.$$

The parameter s controls the eccentricity in the elliptical gaussian envelope, and is usually adapted to the number of orientations by $s = L/4$. If $s = 1$, we see that

$$\psi_\theta(x) = \exp\left\{-\frac{|x|^2}{2\sigma_\psi^2}\right\}\left[\exp\left(i\langle\begin{pmatrix}\xi & 0\end{pmatrix}, r_\theta x\rangle\right) - \exp\left(-\frac{\sigma_\psi^2\xi^2}{2}\right)\right]. \tag{5.6}$$

The parameter $\xi \in \mathbb{R}$ controls the frequency of the oscillatory exponential. The spread of the gaussian envelopes for ϕ and ψ is controlled by σ_ϕ and σ_ψ respectively.

The gaussian low-pass filter is dilated corresponding to a predefined scale 2^J,

$$\phi_{2^J}(x) = 2^{-2J}\phi(2^{-J}x) = \frac{1}{2\pi(2^J\sigma_\phi)^2}e^{-\frac{|x|^2}{2(2^J\sigma_\phi)^2}}. \tag{5.7}$$

The wavelet is both dilated and rotated

$$\psi_\lambda = \psi_{j,\theta}(x) = 2^{-2j}\psi_\theta(2^{-j}x), \quad \lambda = 2^j r_\theta, \tag{5.8}$$

with $\theta = \frac{\pi l}{R}$, for

$$l \in [0, R-1].$$

Note that we only consider positive rotations. In the numerical part we will dilate the wavelet with the scales

$$j \in [0, J-1],$$

so that J is the number of scales and R is the number of rotations. We will refer to the collection of dilated and rotated wavelets together with the dilated low-pass filter as the filter bank.

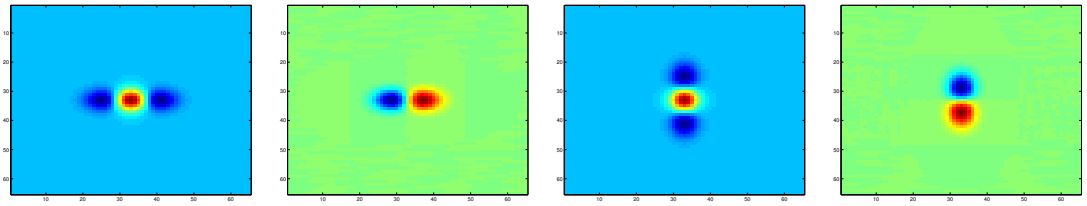
Example 5.1. If $R = 2$, the set of positive rotations G^+ contains two elements r_0 and r_1, where r_0 is a rotation by 0 radians and r_1 a rotation by $\pi/2$ radians. If in addition $J = 2$ we have four different combinations of scales and rotations. Hence

$$\Lambda_J = \{\lambda_1, \lambda_2, \lambda_3, \lambda_4\} = \{2^0 r_0, 2^0 r_1, 2^1 r_0, 2^1 r_1\}.$$

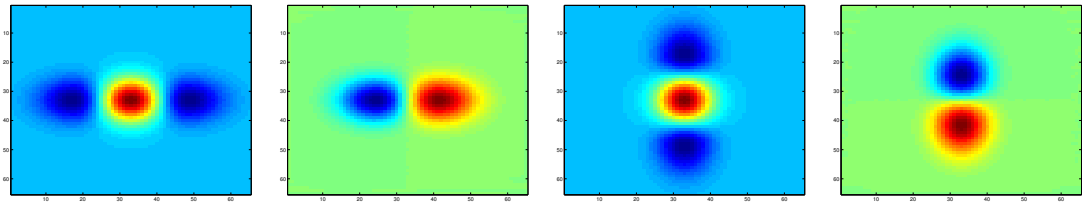
In Figure 5.4 we see the corresponding filter bank with the dilated low-pass filter, and dilated and rotated wavelets.

5.3.2 Scattering coefficients

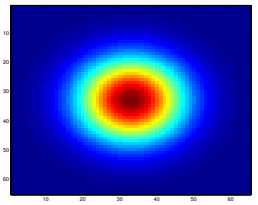
The scattering coefficients are the output from the windowed scattering transform applied to an image f. In Remark 4.3 in Section 4.2 we saw that if we choose M number of layers, and if the set Λ_J contains K elements, then the output of the windowed scattering transform of an image $f \in L^2(\mathbb{R}^2)$, will be a vector of



(a) $\operatorname{Re}\psi_{0,0}$. (b) $\operatorname{Im}\psi_{0,0}$. (c) $\operatorname{Re}\psi_{0,\pi/2}$. (d) $\operatorname{Im}\psi_{0,\pi/2}$.



(e) $\operatorname{Re}\psi_{1,0}$. (f) $\operatorname{Im}\psi_{1,0}$. (g) $\operatorname{Re}\psi_{1,\pi/2}$. (h) $\operatorname{Im}\psi_{1,\pi/2}$.



(i) ϕ_{2^J}.

FIGURE 5.4: Displaying the filterbank, ϕ_{2^J} and $\psi_{j,\theta}$ for $j = 0, 1$ and $\theta = 0, \pi/2$. Here we have used the following parameters: $J = R = 2, \sigma_\phi = \sigma_\psi = 5, \xi = 0.2, s = 2$

.

$N'_J = M \times \sum_{m=0}^{M} K^m$ functions in $L^2(\mathbb{R}^2)$. However as noted in the end of Section 4.2, the energy in the signal is mostly concentrated along frequency-decreasing paths. Therefore our algorithm only computes the scattering coefficients along these paths. We will refer to $\mathcal{P}^M_{J\downarrow}$ as the set of all frequency-decreasing paths of length smaller than or equal to M. The number of such paths will be denoted by $N'_{J\downarrow}$.

Example 5.2. Let Λ_J be the same set as in Example 5.1. The paths of length $m = 2$ will be the set

$$\Lambda_J^2 = \Lambda_J \times \Lambda_J = \{(\lambda_m, \lambda_n) : m, n = 1, 2, 3, 4\},$$

hence $N'_J = 16$. To find the frequency-decreasing paths of length $m = 2$, we note that $|\lambda_1| = |\lambda_2| = 1$ and $|\lambda_3| = |\lambda_4| = 2$, so that the set of frequency-decreasing paths of length $m = 2$ is

$$\Lambda^2_{J\downarrow} = \{(\lambda_1, \lambda_3), (\lambda_1, \lambda_4), (\lambda_2, \lambda_3), (\lambda_2, \lambda_4)\}.$$

Hence

$$\mathcal{P}^2_{J\downarrow} = \Lambda^0_J \cup \Lambda^1_J \cup \Lambda^2_{J\downarrow} = \{\emptyset, \lambda_1, \lambda_2, \lambda_3, \lambda_4, (\lambda_1, \lambda_3), (\lambda_1, \lambda_4), (\lambda_2, \lambda_3), (\lambda_2, \lambda_4)\},$$

and $N'_{J\downarrow} = 8$.

If R is the number of rotations, and J the number of scales in Λ_J, then we can find the number of frequency-decreasing paths by noticing the following:

- For $m = 0$, the output will be only one path, namely $S_J[\emptyset]f$.
- For $m = 1$, every path is frequency-decreasing. Therefore the number of paths are $J \times R$.
- For $m = 2$ there are $\binom{J}{2}$ possible ways to combine frequency-decreasing paths. In addition we have all possible combinations of rotations, that is R^2 ways. In total there are $R^2 \times \binom{J}{2}$ different frequency-decreasing paths for $m = 2$.
- At the k'th layer we see by similar reasoning that the number of frequency-decreasing paths are $R^k \times \binom{J}{k}$.

The total number of frequency-decreasing paths of length smaller than or equal to M is therefore given by

$$N'_{J\downarrow} = \sum_{m=0}^{M} R^m \times \binom{J}{m}. \tag{5.9}$$

Unlike Section 4.2, we now consider digital images. We saw in the introduction of this chapter, that we can view a digitalized image f as an element of $\mathrm{M}_{N,K}(\mathbb{R})$.

When filtering a digital image, the filter is converted into a matrix called the mask. An example of a Gaussian maks of size 3×3 is

$$\frac{1}{16}\begin{pmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{pmatrix}.$$

A convolution is obtained by sliding the mask over the image, and multiplying. In our case, the mask is chosen to be of the size of the input image. However the values in the mask are determined by the scale J. If J is large, the spread of the Gaussian filter is wide, so that the averaging captures more pixels than if J is small. The resulting convolution will therefore have neighbouring pixels of approximately the same size. It is therefore redundant to store all pixels in the convolved image. Instead the resulting image obtained from the convolution is sampled uniformly at intervals of size 2^J. The resulting $N'_{J\downarrow}$ output images

will therefore be images of size $2^{-J}N \times 2^{-J}K$, and can be viewed as elements of $\mathrm{M}_{2^{-J}N,2^{-J}K}(\mathbb{R})$.

If we consider the isomorphism of the two spaces $\mathrm{M}_{2^{-J}N,2^{-J}K}(\mathbb{R}) \cong \mathbb{R}^{2^{-2J}N\times K}$, each output-function can be viewed as a column vector of length $2^{-2J}NK$. Thus the number of scattering coefficients, computed along frequency-decreasing paths of length smaller than or equal to M will be

$$N_J = 2^{-2J}NK \sum_{m=0}^{M} R^m \times \binom{J}{m}. \tag{5.10}$$

Suppose the set Λ_J, and the number of layers M are fixed, and let us denote $S_J f = S_J[\mathcal{P}_{J\downarrow}^M]f$. Then S_J can be viewed as a mapping from $\mathrm{M}_{N,K}(\mathbb{R})$ to $\mathbb{R}^{N_J}$:

$$f = \begin{pmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,K} \\ \vdots & \vdots & . & \vdots \\ a_{N,1} & a_{N,2} & \cdots & a_{N,K} \end{pmatrix} \xrightarrow{S_J} \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_{N_J} \end{pmatrix} = S_J f,$$

with $y_k \in \mathbb{R}$. Note that the windowed scattering transform computes scattering coefficients along frequency-decreasing paths p, at different positions x. This means that $y_k = S_J[p]f(x)$ for a particular value of p and x.

The distance between any to images f and g, computed with the scattering metric can therefore be found by calculating the usual Euclidean distance

$$d_J(f,g) = \left(\sum_{p,x} \left(S_J[p]f(x) - S_J[p]g(x)\right)^2\right)^{1/2} = \sqrt{\sum_{k=1}^{N_J} \left(y_k^f - y_k^g\right)^2}.$$

Here y_k^f and y_k^g are the scattering coefficients of f and g respectively.

Example 5.3. If f is an image of $N \times K = 256 \times 256$ pixels, and $J = R = 2$, then $N' = K' = 2^{-2} \cdot 256 = 64$. Moreover, if $M = 2$, the number of scattering coefficients would be

$$\begin{aligned} N_J &= 2^{-4} \cdot 256 \cdot 256 \sum_{m=0}^{2} 2^m \times \binom{2}{m} \\ &= 2^{-4} \cdot 256 \cdot 256 \left(1 \times \binom{2}{0} + 2^1 \times \binom{2}{1} + 2^2 \times \binom{2}{2}\right) = 36,864. \end{aligned}$$

5.3.3 Energy propagation

We are interested in finding out how many layers we need to include in order not to lose too much information about the images. This can be done by investigating how the energy in the signals propagates through the layers. We can compute the percentage of the scattering energy, $\|S_J[\Lambda_{J\downarrow}^M]f\|/\|f\|$, captured by frequency-decreasing paths of length m as a function of J. To do this we will use the Shannon wavelet defined in Example 3.7, which preserves the energy in the signal. In Table 5.1 we have computed the energy for different scales at different layers

J	$m=0$	$m=1$	$m=2$	$m=3$	$m=4$	$m\leq 3$
1	99.0785	0.9039	-	-	-	99.9824
2	96.8725	2.5773	0.0320	-	-	99.4818
3	92.9176	6.2447	0.2464	0.0027	-	99.4114
4	87.5740	10.9963	0.7400	0.0237	0.0003	99.3340
5	82.4112	15.2211	1.5352	0.0836	0.0024	99.2510
6	77.9784	18.2728	2.7210	0.1969	0.0088	99.1691
7	74.2455	20.8642	3.6274	0.3384	0.0202	99.0756
8	52.0543	38.5836	7.4342	0.9494	0.0711	99.0216

TABLE 5.1: Percentage of scattering energy $\|S_J[\Lambda^M_{J\downarrow}]f\|/\|f\|$ captured by frequency-decreasing paths of length m as a function of J. The values are computed on US images with Shannon wavelets.

J	$m=0$	$m=1$	$m=2$	$m=3$	$m=4$	$m\leq 3$
1	96.686	0.208	-	-	-	96.894
2	93.155	0.382	0.005	-	-	93.542
3	88.285	0.592	0.010	$8.76{\cdot}10^{-5}$	-	88.887
4	83.463	0.788	0.017	$2.16{\cdot}10^{-4}$	$1.34{\cdot}10^{-6}$	84.268
5	79.403	0.980	0.024	$3.92{\cdot}10^{-4}$	$3.81{\cdot}10^{-6}$	80.407
6	75.718	1.185	0.031	$6.05{\cdot}10^{-4}$	$8.11{\cdot}10^{-6}$	76.934
7	72.953	1.409	0.037	$8.21{\cdot}10^{-4}$	$1.42{\cdot}10^{-5}$	74.399
8	65.900	1.498	0.042	$1.01{\cdot}10^{-3}$	$2.06{\cdot}10^{-5}$	67.440

TABLE 5.2: Percentage of scattering energy $\|S_J[\Lambda^M_{J\downarrow}]f\|/\|f\|$ captured by frequency-decreasing paths of length m as a function of J. The values are computed on US images with Morlet wavelets, with $R=4$.

using the Shannon wavelet. We see that for $J \leq 8$, more than 99% of the energy is contained in the three first layers. The results from Table 5.1 tell us three things. First, limiting the computation to frequency-decreasing paths is a good approximation, since we only lose under one percent of the energy. Second, if we want to compute the windowed scattering transform of images of size $2^8 \times 2^8$ then choosing three layers will give a loss of information by under 1%. Finally we see that when the scale J increases, more of the energy propagates towards deeper layers. Hence, if we choose a large scale J, we might need to include more layers in order not to lose vital information.

The percentage of the scattering energy computed with the Morlet wavelet is provided in Table 5.2. This table clearly indicates that energy is not conserved by the windowed scattering transform when using this wavelet. For $m = 0$, that is after filtering the image with the scaling function ϕ, the amount of energy is about the same for both the Shannon wavelet and the Morlet wavelet. When $m = 1$, we have applied the wavelet transform, the modulus operator, and a filtering. In this column, there is a huge difference in the two tables. The reason is that the operator W_J is not unitary in the case when ψ is the Morlet wavelet. Hence, energy is lost when the wavelet transform is applied.

5.3.4 Visualization of the scattering coefficients

To compute the windowed scattering transform we will iteratively apply the wavelet transform and modulus operator. The output is then filtered with a low-pass filter. The first iterations of this procedure applied to the image in Figure 5.5(a), with Λ_J as in Example 5.1, are shown in Figure 5.5(b)-5.5(f). Here we see the convolution with the real and imaginary parts of the wavelet ψ, dilated and rotated according to $\lambda_1 = 2^0 r_0$.

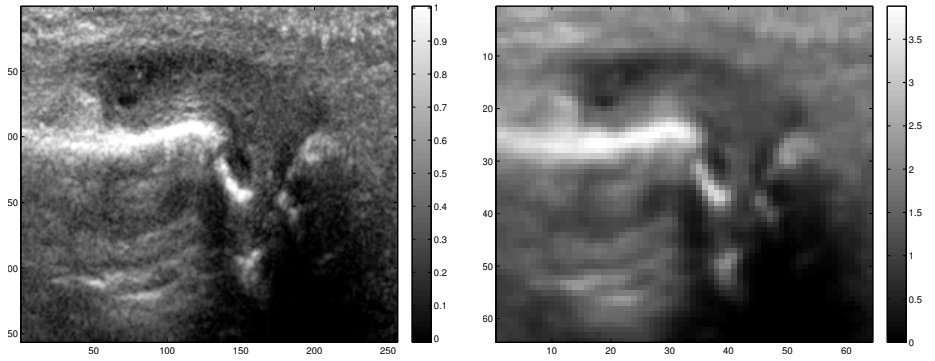
The scattering coefficients from the first layer, that is paths of length 1, are displayed in Figure 5.6.

The scattering coeffcients in the second layer, computed along the frequency-decreasing paths are illustrated in Figure 5.7. These are the same paths as we found in Example 5.2.

We can also display the scattering coefficients for a fixed position x in the image. Let $\{\Omega[p]\}_{p\in\Lambda_J^m}$ be a partition of $\mathbb{R}^2$, so that for each $\omega \in \mathbb{R}^2$ we associate a path $p(\omega)$. In other words, each $\omega \in \mathbb{R}^2$ belongs to exactly one of the sets $\Omega[p]$. This partition is illustrated in Figure 5.8 for $J = R = 6$, and $m = 1$ and $m = 2$. Each annular sector corresponds to the value of $S_J[p]f(x)$ for some path p, so that $S_J[p(\omega)]f(x)$ is a piecewise constant function of ω. In Figure 5.8(a) the partition is shown for $m = 1$. The annular sector highlighted in blue corresponds to the path $p = \lambda = 2^{j_4} r_5$. This annular sector approximates the frequency support of $\widehat{\psi}_{2^{j_4} r_5}$, and the size of this annular sector is proportional to $\|\psi_{2^{j_4} r_5}\|^2$.

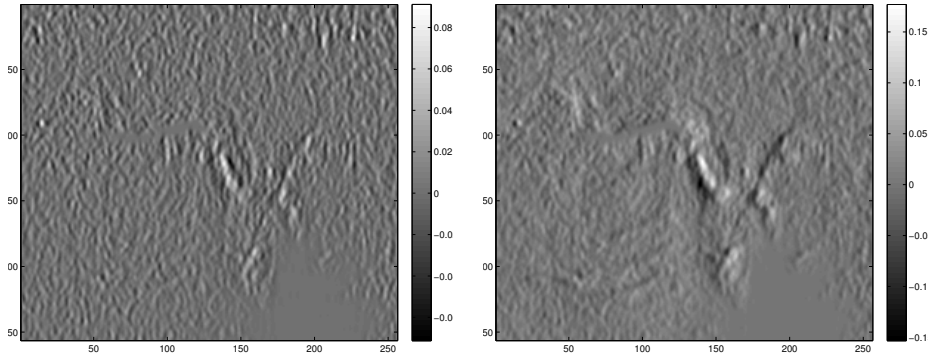
In Figure 5.8(b) we have applied an additional layer, where the boundaries from the first layer is highlighted in red. Each annular sector $\Omega[2^j r]$ is subdivided along the radial axis, and along the angular axis. Since we only consider frequency decreasing paths the angular sectors furthest away from the origin is subdivided more in the radial direction, than the angular sectors closer to the origin. In Figure 5.8(b) we have marked the domain $\Omega[2^{j_1} r_2, 2^{j_2} r_1]$, whose size is proportional to $\||\psi_{2^{j_1} r_2}| * \psi_{2^{j_2} r_1}\|^2$.

The scattering coefficients of the image in Figure 5.5(a), with 256×256 pixels, computed with $J = 6$ will be images of 4×4 pixels. In Figure 5.9 we display these scattering coefficients, where each subdivided disc corresponds to a pixel.



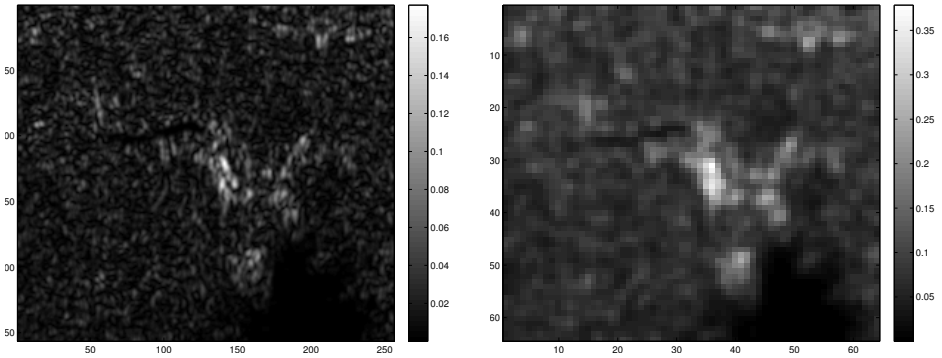
(a) Original image f.

(b) $f * \phi_J = S_J[\emptyset]f$.



(c) $f * \operatorname{Re}\psi_{\lambda_1}$.

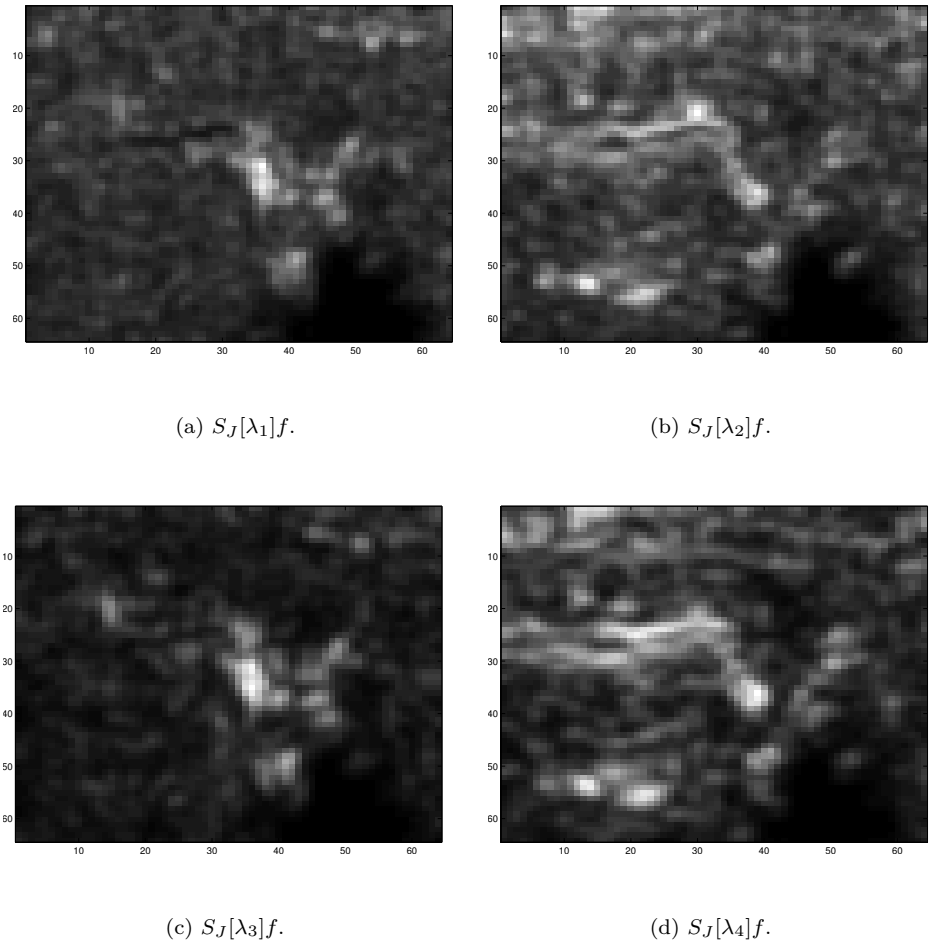
(d) $f * \operatorname{Im}\psi_{\lambda_1}$.



(e) $|f * \psi_{\lambda_1}|$.

(f) $|f * \psi_{\lambda_1}| * \phi_J = S_J[\lambda_1]f$.

FIGURE 5.5: Illustrates the first iterations of the windowed scattering transform.



(a) $S_J[\lambda_1]f$. (b) $S_J[\lambda_2]f$.

(c) $S_J[\lambda_3]f$. (d) $S_J[\lambda_4]f$.

FIGURE 5.6: Illustrates the scattering coefficients from the first layer.

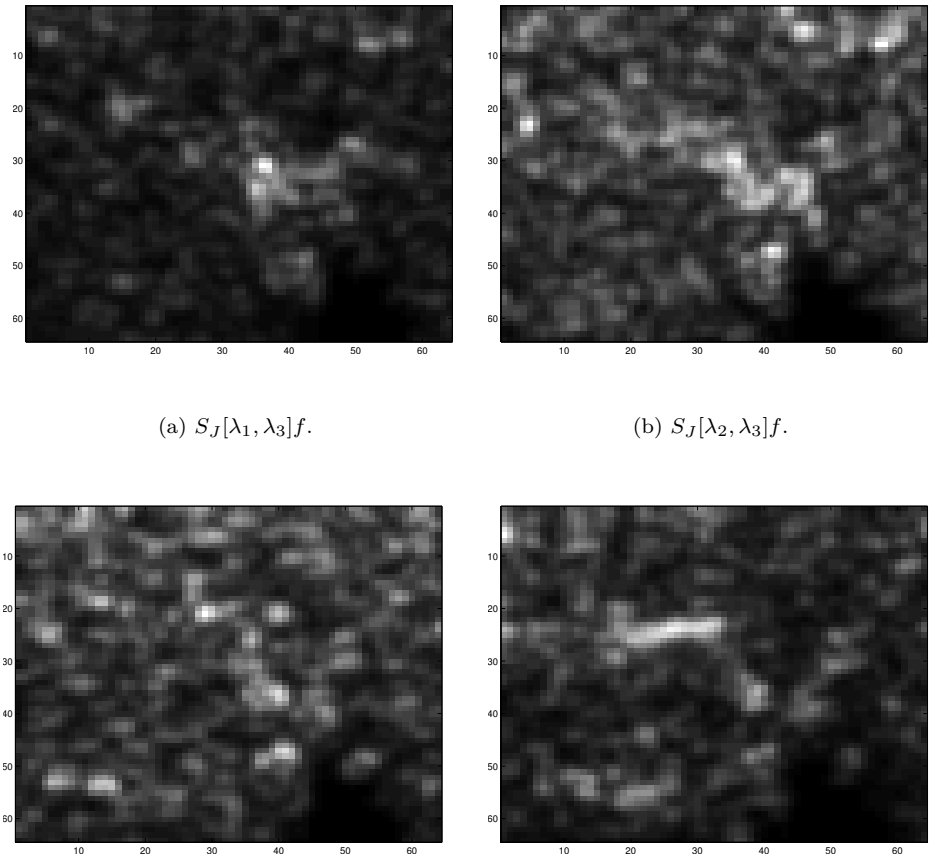
5.4 Classification

In this section we will explain how we can classify images based on the scattering coefficients and the area of inflammation. We will assume that all images have a discrete representation, i.e that if f is an image, then f can be represented as a matrix of size $N \times K$.

A typical problem when dealing with classification is the following: Given T classes, $\mathcal{C}_1, \mathcal{C}_2, \cdots, \mathcal{C}_T$, and the training set

$$X = \left\{ \{f_1^1, f_1^2, \cdots f_1^{L_1}\}, \{f_2^1, f_2^2, \cdots f_2^{L_2}\}, \cdots, \{f_T^1, f_T^2, \cdots f_T^{L_T}\} \right\},$$

with $f_i^l \in \mathcal{C}_i$, use this training set to find a method, so that given a new signal g we are able to determine which class it belongs to. The function which assigns a new signal to a class based on a certain method, will be referred to as the classifier. Our signals f_i^l are US images of finger-joints, where four classes represent different degrees of inflammation. We will first present some basic statistical theory which will be needed later in this section.



(a) $S_J[\lambda_1, \lambda_3]f$.

(b) $S_J[\lambda_2, \lambda_3]f$.

(c) $S_J[\lambda_1, \lambda_4]f$.

(d) $S_J[\lambda_2, \lambda_4]f$.

FIGURE 5.7: Illustrates the scattering coefficients from the second layer.

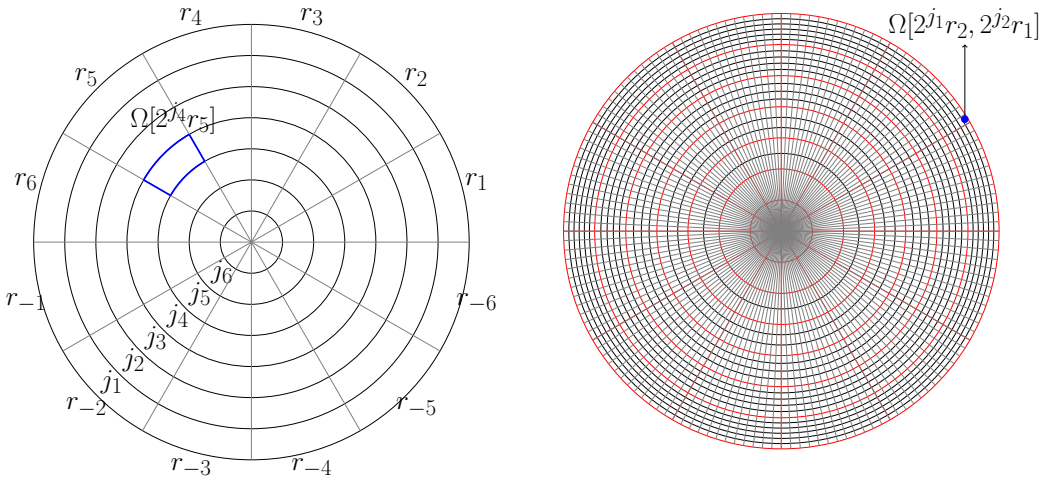
5.4.1 Statistical background

We shall represent each class $\mathcal{C}_i$ by a stochastic process F_i. A stochastic process F_i is a family $\{F_i(x) : x \in \mathcal{I}\}$, where $\mathcal{I}$ is some index set. In our case $\mathcal{I} = \{(1,1),(1,2),\cdots,(N,K)\}$, where each element in $\mathcal{I}$ corresponds to a pixel in an image. From each class we will have a collection of training images $f_i^l \in \mathcal{C}_i$. We say that these images are realizations or observations of the process F_i. If there are L_i realizations of F_i, then for a given point x, an estimate for the expected value $E[F_i(x)] = \mu_{i,x}$ may be found from the empirical average

$$\widetilde{\mu}_{i,x} = \frac{1}{L_i} \sum_{l=1}^{L_i} f_i^l(x).$$

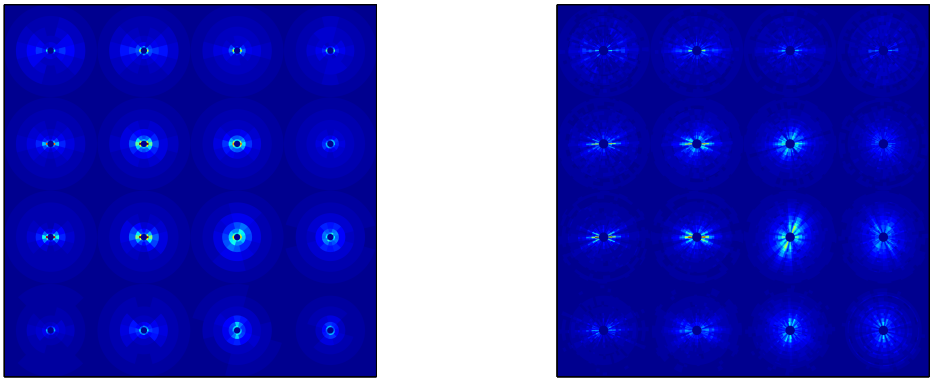
Similarly an estimate for the variance $\mathrm{Var}(F_i(x)) = \Sigma_{i,x}$ is given by the empirical variance

$$\widetilde{\Sigma}_{i,x} = \frac{1}{L_i - 1} \sum_{l=1}^{L_i} (f_i^l(x) - \widetilde{\mu}_{i,x})^2.$$



(a) $\{\Omega[p]\}_{p\in\Lambda_J^1}$. (b) $\{\Omega[p]\}_{p\in\Lambda_J^2}$.

FIGURE 5.8: Partition of $\mathbb{R}^2$ in angular sectors, with each angular sector corresponding to a path $p \in \Lambda_J^m$.



(a) $m = 1$. (b) $m = 2$.

FIGURE 5.9: The scattering coefficients of the image in Figure 5.5(a) with $J = R = 6$. The left figure shows the coefficients with $m = 1$ and the right for $m = 2$.

Here $\tilde{\cdot}$ means that the parameter is an unbiased estimator. If f_i^l is arranged in a column vector

$$f_i^l = \begin{pmatrix} f_i^l(1,1) \\ f_i^l(1,2) \\ \vdots \\ f_i^l(N,K) \end{pmatrix},$$

with each coordinate corresponding to a point $x \in \mathcal{I}$, then the empirical average of F_i may be written as

$$\widetilde{\mu}_i = \frac{1}{L_i}\sum_{l=1}^{L_i} f_i^l = \begin{pmatrix} \frac{1}{L_i}\sum_{l=1}^{L_i} f_i^l(1,1) \\ \frac{1}{L_i}\sum_{l=1}^{L_i} f_i^l(1,2) \\ \vdots \\ \frac{1}{L_i}\sum_{l=1}^{L_i} f_i^l(N,K). \end{pmatrix}.$$

Accordingly, the empirical variance-covariance matrix is

$$\widetilde{\Sigma}_i = \frac{1}{L_i - 1}\sum_{l=1}^{L_i}(f_i^l - \widetilde{\mu}_i)(f_i^l - \widetilde{\mu}_i)^T.$$

Operators, like the scattering operator S_J can also be applied to this process. An estimate for the expected scattering coefficients can then be found by first applying the operator S_J to all realizations, and then carrying out the procedure described above.

5.4.2 Classification from area of inflammation

We want to classify images based on the area of inflammation. For each image $f_i^l \in \mathcal{C}_i$ there is a corresponding area A_i^l. That is, if $f_i^l \in \mathcal{C}_i$ and $\mathcal{A}$ is an operator $\mathcal{A} : L^2(\mathbb{R}^2) \to \mathbb{R}^+$ which takes US-images as input, and output the area of inflammation, then $A_i^l = \mathcal{A}f_i^l$.

The process F_i will have an expected area of inflammation $E[\mathcal{A}F_i] = \mu_i^{\mathcal{A}}$. The expected area can be estimated by the empirical average

$$\widetilde{\mu}_i^{\mathcal{A}} = \frac{1}{L_i}\sum_{l=1}^{L_i}\mathcal{A}f_i^l = \frac{1}{L_i}\sum_{l=1}^{L_i} A_i^l. \tag{5.11}$$

The variance $\mathrm{Var}(\mathcal{A}F_i) = \Sigma_i^{\mathcal{A}}$, can be estimated by

$$\widetilde{\Sigma}_i^{\mathcal{A}} = \frac{1}{L_i - 1}\sum_{l=1}^{L_i}(A_i^l - \widetilde{\mu}_i^{\mathcal{A}})^2. \tag{5.12}$$

From these estimates we can build a prediction interval associated with each class. A prediction interval is an interval (a_i, b_i) centred around the empirical average $\mu_i^{\mathcal{A}}$, and is used to predict future observations. Before constructing a prediction interval a parameter α needs to be chosen. This parameter controls the probability for a future observation to be in the prediction interval. We assume

that the process $\mathcal{A}F_i$ is normal distributed with mean $\mu_i^{\mathcal{A}}$ and variance $\Sigma_i^{\mathcal{A}}$. In that case a $(1-\alpha)$ prediction interval is given by [EC12, p.136]

$$(a_i, b_i) = \left(\widetilde{\mu}_i^{\mathcal{A}} - T_{1-\alpha/2}\sqrt{\Sigma_i^{\mathcal{A}}\left(1+\frac{1}{L_i}\right)}, \widetilde{\mu}_i^{\mathcal{A}} + T_{1-\alpha/2}\sqrt{\Sigma_i^{\mathcal{A}}\left(1+\frac{1}{L_i}\right)}\right),$$

where $T_{1-\alpha/2}$ is the $100\left(1-\frac{\alpha}{2}\right)$th percentile of Student's t-distribution with $L_i - 1$ degrees of freedom. If f is a future observation then $\text{Prob}(f \in (a_i, b_i)) = 1-\alpha$.

5.4.2.1 Classifier

Let $\{(a_i, b_i)\}_{i=0,1,2,3}$ denote the prediction intervals for the classes $\mathcal{C}_i, i \in \{0,1,2,3\}$. The index i corresponds to the class of US images with degree of inflammation i. A new image f, with corresponding area of inflammation A, is classified according to the shortest distance to the prediction intervals. In other words, f is classified as class $\mathcal{C}_\iota$, where ι is given by

$$\iota = \underset{i\in\{0,1,2,3\}}{\text{argmin}}\ \min\{|A-a_i|, |A-b_i|\}. \tag{5.13}$$

As each class will be associated with a prediction interval, it is important that all intervals are mutually disjoint. If two intervals have a non-empty intersection we will have problems when classifying an image which lies in the intersection. The size of the intervals can therefore be controlled by the parameter α so that the intervals are mutually disjoint. This parameter must however be the same for all intervals, otherwise we will favor intervals with a smaller value of α.

5.4.3 Classification from scattering coefficients

We want to classify images based on their scattering coefficients. From the realizations of the stochastic process F_i, we can estimate the expected scattering coefficients $\text{E}[S_J F_i] = \mu_i$ and the variance-covariance matrix $\text{Var}(S_J F_i) = \Sigma_i$. For a given realization f_i^l of the process F_i, we saw in Section 5.3.2 that the coefficients $S_J f_i^l$ belong to the space $\mathbb{R}^{N_J}$. After computing the scattering coefficients of all the realizations of a given class, we look for lower dimensional approximations which best describe the signals in each class. This is achieved by performing a principal component analysis(PCA). The output will be an affine space, approximating the scattering coefficients of each stochastic process F_i. A new signal will be classified according to the closest affine space. Since each class $\mathcal{C}_i$ is processed individually we will fix a class $\mathcal{C}$ and the corresponding stochastic process F.

5.4.3.1 Choice of scale

It is important that an appropriate value for the scale variable J is chosen. According to Lemma 4.10, the scattering distance decreases as J increases. Increasing J will increase the translation invariance, but there is a loss of information that causes different signals to be closer together. This may cause misclassifications.

The optimal scale J should therefore be determined by the maximum pixel displacement due to translation. In some cases the optimal scale J is chosen so that 2^J equals the image width. In our case an analysis of the optimal scale will be carried out in Section 6.3.2.3, after we have presented the results.

5.4.3.2 PCA

Although the theory behind PCA is covered in any introductory book on multivariate statistics we will outline the main idea in view of the scattering coefficients. The theory is taken from [EC12].

Let $\mathcal{Y} = \{S_J f_1, S_J f_2, \cdots, S_J f_L\}$ denote the training set of scattering coefficients of L realizations of the process F. This means that $S_J f_l = \left(y_1^l, y_2^l, \cdots, y_{N_J}^l\right)$ for some signal f_l belonging to the class $\mathcal{C}$. Here y_i^l denotes scattering coefficient number i for signal number l. These coefficients can be arranged in an $L \times N_J$ matrix which we denote by $\mathbf{Y}$:

$$\mathbf{Y} = \begin{pmatrix} y_1^1 & y_2^1 & \cdots & y_{N_J}^1 \\ y_1^2 & y_2^2 & \cdots & y_{N_J}^2 \\ \vdots & \vdots & \cdots & \vdots \\ y_1^L & y_2^L & \cdots & y_{N_J}^L \end{pmatrix} = \begin{pmatrix} Y_1^T \\ \vdots \\ Y_L^T \end{pmatrix} = \begin{pmatrix} S_J F_1 & \cdots & S_J F_{N_J} \end{pmatrix}. \tag{5.14}$$

Here Y_l is a column vector of length N_J corresponding to the l'th row in $\mathbf{Y}$, and $S_J F_n$ is a column vector of length L corresponding to the n'th column in $\mathbf{Y}$. Note that each $S_J F_n$ is a vector of L observations of the same coefficient. If $E[S_J F_n] = \mu_n$ is the expected value for the n'th coefficient of the stochastic process F, then it can be estimated by the empirical average

$$\widetilde{\mu}_n = \frac{1}{L} \sum_{l=1}^{L} y_n^l.$$

Let $E[S_J F] = E[\{S_J[p]F(x)\}_{p,x}]$ be the collection of all expected scattering coefficients of the stochastic process F. Then an unbiased estimator for these coefficients is

$$\widetilde{\mu} = \begin{pmatrix} \widetilde{\mu}_1 \\ \widetilde{\mu}_2 \\ \vdots \\ \widetilde{\mu}_{N_J} \end{pmatrix}.$$

To perform a PCA we need an estimate for the variance-covariance matrix Σ. This is given by the empirical variance-covariance matrix:

$$\widetilde{\Sigma} = \frac{1}{L} \sum_{n=l}^{L} (Y_l - \widetilde{\mu})(Y_l - \widetilde{\mu})^T. \tag{5.15}$$

This will be an $N_J \times N_J$ matrix. Let $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_{N_J}$ be the eigenvalues of $\widetilde{\Sigma}$ arranged in a descending order, and let $p_1, \cdots, p_{N_J}$ be the corresponding eigenvectors. We will refer to the i'th *principal axis* of $\mathbf{Y}$ as the direction of

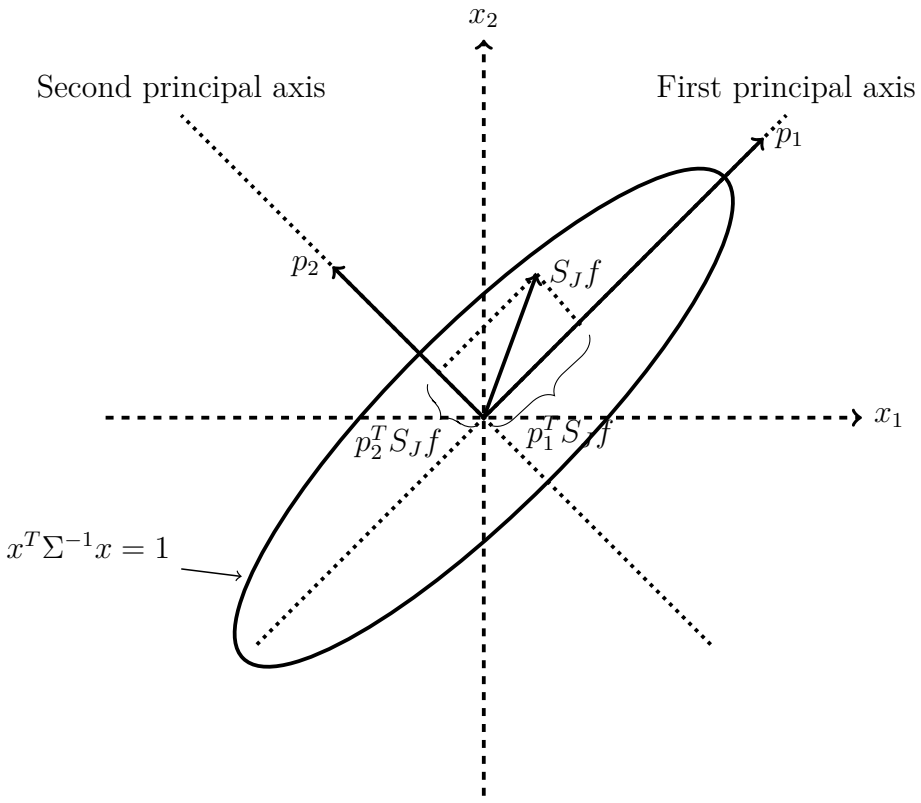


FIGURE 5.10: Principal component analysis for two-dimensional scattering coefficients.

the eigenvector p_i corresponding to the i'th largest eigenvalue. The i'th *principal component* of $\mathbf{Y}$ is the projection of $\mathbf{Y}$ onto the i'th principal axis. The matrix $\mathbf{Y}$ can be represented in terms of its principal components by a change of basis from the canonical basis vectors to the eigenvectors of the variance covariance matrix. If the eigenvectors p_i are arranged as columns in a matrix $\mathbf{P}$, then this change of basis is $\mathbf{Y}_P = \mathbf{P}^T\mathbf{Y}$.

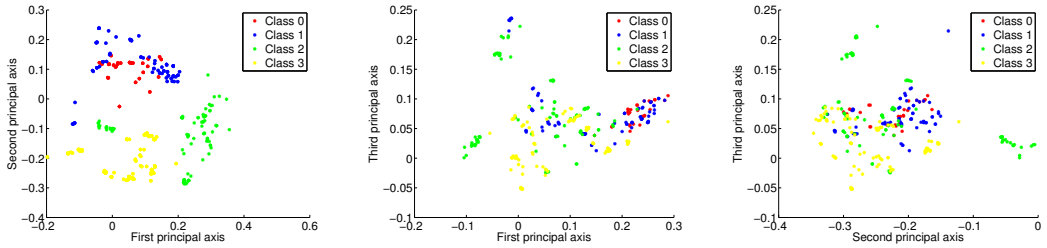
Example 5.4. Let us for simplicity assume that $N_J = 2$, so that the scattering coefficients of each realization $S_J f$, can be viewed as a vector in $\mathbb{R}^2$. Let p_1 and p_2 denote the eigenvectors of the variance-covariance matrix. The projection of a signal $S_J f$ onto the principal axes, is illustrated in Figure 5.10. The ellipse given by the equation $x^T\Sigma^{-1}x = 1$, contains all realizations in its interior, and has its main axes along the principal axes. The first principal axis corresponding to the largest eigenvalue, lies along the major axis, while the second principal axis corresponding to the smallest eigenvalue, lies along the minor axis.

This also generalizes to higher dimensions, that is the major axes of the ellipsoid which contains all realizations of the process $S_J F$ are the first principal axes. For scatterings coefficients belonging to $\mathbb{R}^{N_J}$ we may choose the linear space spanned by the d first principal axes to approximate these coefficients.

Example 5.4 illustrates why projections on the principal axes will give a good description of the signals in the training set. Among all linear spaces of dimension d, the space spanned by the first d eigenvectors of the variance-covariance matrix,

N	1	2	3	5	10	20
Class 0	0.3617	0.5234	0.6721	0.8432	0.9461	0.9990
Class 1	0.4763	0.7145	0.7810	0.8747	0.9495	0.9879
Class 2	0.4931	0.7353	0.8094	0.8944	0.9633	0.9931
Class 3	0.4895	0.6464	0.7549	0.8521	0.9553	0.9936

TABLE 5.3: Size of the first eigenvalues in Σ compared to all the eigenvalues, $\sum_{k=1}^{N} \lambda_k / \sum_{k=1}^{N_J} \lambda_k$. The eigenvalues are computed from the scattering coefficients of 296 images with $J = 8$ and $R = 2$.



(a) First and second principal components.

(b) First and third principal components.

(c) Second and third principal components.

FIGURE 5.11: Illustrates the scattering coefficients computed with $J = 8$ and $R = 2$, projected onto the first, second and third principal axes.

will minimize the expected quadratic error [Bru12]. If the eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_{N_J}$ are arranged in a descending order, then the variation in the coefficients explained by the m'th principal component is given by the fraction $\lambda_m / \sum_{k=1}^{N_J} \lambda_k$ [EC12, p.291]. Hence, if the eigenvalues become small very fast, then only a few components are necessary to get a good approximation. However if all the eigenvalues are of the same size, then we may have to include almost every principal component to get a descent approximation. In Table 5.3 we have computed the fraction $\sum_{k=1}^{N} \lambda_k / \sum_{k=1}^{N_J} \lambda_k$ for different values of N. The scattering coefficients are computed with $J = 8$, $R = 2$ and $M = 3$, which means that $N_J = 577$ for each class.

This table shows that projecting the coefficients onto the first three principal axes will explain $67 - 80\%$ of the total variation in the coefficients, depending on the class. The 20 first principal axes will explain about 99% of the variation.

In Figure 5.11 we see these coefficients projected onto their first, second and third principal axes. The different colors represent the different classes of images. These figures show that there is a clustering of images which belong to the same class.

5.4.3.3 Affine space from PCA

From the PCA one can build an affine space which approximates the scattering coefficients in each class.

Consider a class $\mathcal{C}$, and let $E[S_J F]$ denote the expected scattering coefficients of the stochastic process F. Let further $\mathbb{V}_d$ be the linear space spanned by the d

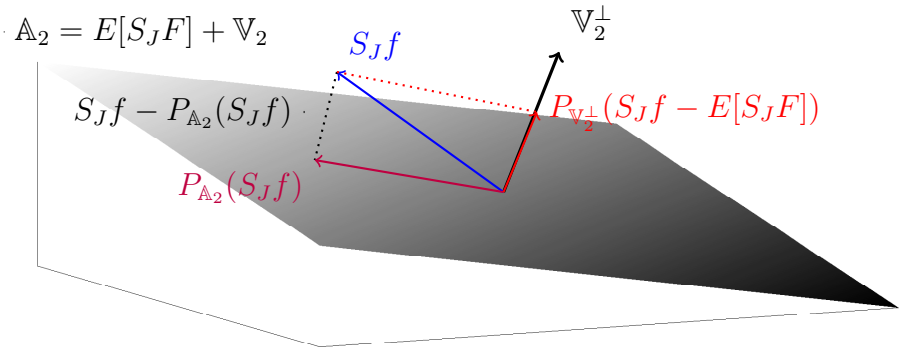


FIGURE 5.12: Two-dimensional scattering coefficients projected onto an affine space $\mathbb{A}_2 = E[S_J F] + \mathbb{V}_2$, and its orthogonal complement.

first principal axes. Then the affine space

$$\mathbb{A}_d = E[S_J F] + \mathbb{V}_d,$$

will be used as an approximation of the scattering coefficients of signals belonging to this class. We will have four classes $\mathcal{C}_i$ for $i = 0, 1, 2, 3$, each representing a different degree of inflammation. Each such class arises from a separate stochastic process F_i, and its scattering coefficients will be approximated by the affine spaces

$$\mathbb{A}_{d,i} = E[S_J F_i] + \mathbb{V}_{d,i}, \quad k = 0, 1, 2, 3.$$

5.4.3.4 Classifier

Once we have build an affine space for each class we can define the classifier. For an image $f \in L^2(\mathbb{R}^2)$, let $P_{\mathbb{A}_{d,i}}(S_J f)$ be the projection of the scattering coefficients $S_J f$ onto the affine space $\mathbb{A}_{d,i}$. The signal f will be classified according to the smallest distance from the coefficients $S_J f$ to the affine space $\mathbb{A}_{d,i}$. In other words, f will be classified as class C_ι, where ι is given by

$$\iota = \underset{i \in \{0,1,2,3\}}{\operatorname{argmin}} \|S_J f - P_{\mathbb{A}_{d,i}}(S_J f)\|. \tag{5.16}$$

It is easily verified by elementary linear algebra that

$$\|S_J f - P_{\mathbb{A}_{d,i}}(S_J f)\| = \|P_{\mathbb{V}_{d,i}^\perp}(S_J f - E[S_J F_i])\|,$$

where $\mathbb{V}_{d,i}^\perp$ is the orthogonal complement of $\mathbb{V}_{d,i}$, see Figure 5.12. Thus finding the minimal distances to the affine spaces is equivalent to finding the closest centroid centred at $E[S_J F_i]$, without taking the first d principal directions into account. As a result, the classifier therefore keeps the eigenvectors corresponding to the smallest eigenvalues.

5.4.3.5 Overfitting and underfitting

The affine space approximation is a good approximation if most of the variability within each class is described by the d first principal components. In this case $S_J F_i - E[S_J F_i]$ is well approximated by a projection on a low-dimensional space. If d is chosen too small then much of the variability will not be captured by the affine space approximation, and may cause misclassification. This is called underfitting.

If d is chosen too large, random error will be prominent in the model, and we have overfitting. This will typically happen if $d > L_i$, where L_i is the number of training images for the class $\mathcal{C}_i$. This corresponds to the general overfitting dichotomy which appears in the process of image classification. By increasing the number of parameters you may lose robustness of the method.

To find the optimal value for d, a cross-validation procedure can be used. This procedure divides the training set into two separate sets, where one set is used to build the model, and the second is used to test the model for different values for d.

Chapter 6

Results

In this chapter we will present the results from the classification methods described in the previous chapter. An analysis of the results will also be carried out. First we will present the database of images which are available for us to test the different methods.

6.1 Database of images

6.1.1 Data

The images are provided from a research project called Medusa. Medusa is a research project whose aim is to develop a software for identifying and classifying synovitis. The total dataset contains 296 US images of finger joints, where each image has been analyzed and classified by a medical doctor. Images are classified according to the degree of inflammation. A higher degree of inflammation, means that the inflammation is more severe. The distribution of images based on the degree of inflammation is provided in Table 6.1.

All images have also been annotated manually by different persons, so that for each image there is a corresponding image with annotation. These annotated images will be used when calculating the area. The scattering coefficients will be computed from the original images, without annotation.

The provided images are of size 500×840 pixels. To reduce complexity and memory usage, we have reduced the size of the images to 256×256 pixels when computing the scattering coefficients. This has been done by using a bicubic interpolation, where each pixel value is a weighted average of pixels in the nearest 4-by-4 neighborhood. The area is calculated on images with the original size.

Degree 0	Degree 1	Degree 2	Degree 3	Total number of images
29	78	92	97	296

TABLE 6.1: Number of images with different degree of inflammation.

6.1.2 Sources of error

All images are classified by the same medical doctor, which rules out any error concerning different doctors subjective opinion. However, there exist border cases for which the classification given by a medical doctor may differ from that given by the algorithm. Such discrepancies do not really influence the efficiency of the algorithm. These border cases are illustrated in Figure 6.2.

There is a high uncertainty related to the manual annotation of the inflamed region. As two persons may have different opinions on where the inflammation is located, there may be huge differences in the annotation of two similar images. Also the annotation is not verified by the medical doctor who classified the images. Therefore there may be a mismatch between what is recognized as inflammation between the people who have annotated the images, and the medical doctor who has classified them.

Another source of error related to the calculation of the area, is that the images are taken of different patients. The scale of all images are the same, but the finger-size may vary from patient to patient. A patient may in reality have low degree of inflammation, but due to a large finger-size, the area is proportionally large. As a result, the patient will be classified with a higher degree of inflammation if the classification is only based the area.

6.1.3 General setup

The database is divided into a training set and a test set. A model is built out of images from the training set. This model is tested with the classifier on images from the test. This will output an error which is given by

$$\text{Error} = \frac{\text{Number of misclassifications}}{\text{Total number of images in the test set}}.$$

A diagram of this general procedure is provided in Figure 6.1.

6.2 Classification from area

In this section we will present the results from the classification using the area of inflammation.

6.2.1 Setup

The area is calculated from annotated images. This annotation is detected by the Canny edge-detection algorithm. This algorithm is a built-in function in Matlab, and Matlab also has a routine for finding the optimal values for $T_{\text{high}}, T_{\text{low}}$ and s (see Section 3.2 for definitions of these parameters). The area is then calculated by using Greens formula (5.3). Each pixel represents a unit area. A normalization of the area is done by dividing by the total area of the images. The total area of each image is $500 \times 840 = 420.000$.

For each class we build a prediction interval from the calculated areas in the training set. The paramter α is chosen so that the intervals are mutually disjoint.

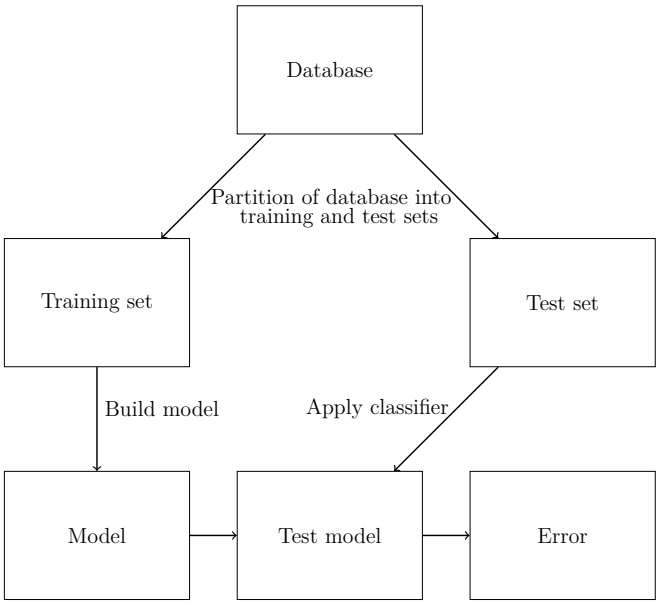


FIGURE 6.1: Diagram of the general setup.

Parameter	Degree 0	Degree 1	Degree 2	Degree 3
$\widetilde{\mu}^{\mathcal{A}}$	0.014127	0.026763	0.053453	0.080391
$\widetilde{\Sigma}^{\mathcal{A}}$	3.37×10^{-5}	1.74×10^{-4}	5.19×10^{-4}	1.40×10^{-3}

TABLE 6.2: Estimated mean and variance from the area of inflammation.

An image in the test set is then classified according to the shortest distance to these intervals. The collection of these intervals together with the classifier will be referred to as the model.

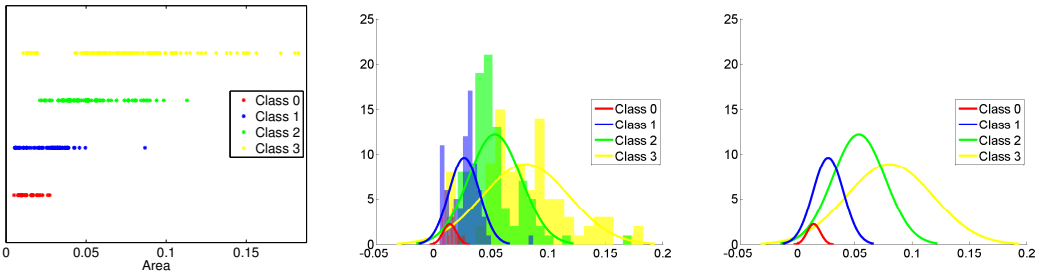
6.2.2 Results from area classifier

We will first see the results, when all images are in both the training set and the test set. These results will be used to highlight weaknesses in the model. Thereafter, the database is partitioned into a training set and a test set, with different ratios between the sizes. At the end we will see the results when the test size is kept fixed, and we vary the training size.

6.2.2.1 Results when using all images

The calculated area for each image is visualized in Figure 6.2(a). The different colors represent the different classes. In Figure 6.2(b) we see a histogram of the estimated area of images within each class. These histograms are fitted with a density function of a normal distribution. These distribution functions are plotted independently in Figure 6.2(c).

The average area $\widetilde{\mu}^{\mathcal{A}}$, and the empirical variance $\widetilde{\Sigma}^{\mathcal{A}}$ for each class are provided in Table 6.2. This table shows that a higher degree of inflammation has a larger

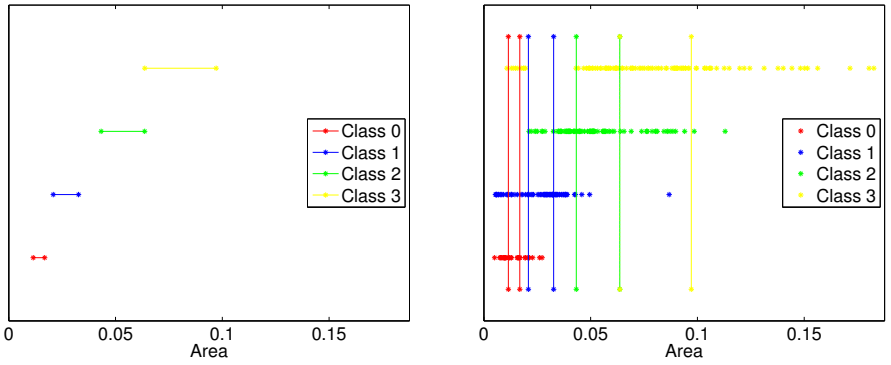


(a) Area for all images.

(b) Histogram of area for all images.

(c) Normal distribution for each class.

FIGURE 6.2: Illustration of the area of all images corresponding to each class.



(a) Prediction intervals.

(b) Prediction intervals and the area of all images.

FIGURE 6.3: Predication intervals estimated from all images in the database.

average area, which confirms the hypothesis that the area of the inflamed region is dependent on the degree of inflammation.

From these values one can build a prediction interval for each class. The resulting prediction intervals are illustrated in Figure 6.3(a). These intervals are chosen as large as possible so that they are non-overlapping and have the same probability to contain a future observation. The probability is controlled by the parameter α which in this case happens to be 0.6592. This means that for each interval, the probability for a future observation to lie in the correct interval is 34.08%. A higher value for α would give a non-empty intersection for the intervals for class 3 and 2. An illustration of the predication intervals, drawn as vertical lines, together with the area for each class is provided in Figure 6.3(b). From this figure it is clear that there are already images which are misclassified. This is due to the high variance within each class. Figure 6.4 shows the amount of correct and wrong classifications within each class. In total 61.5% of all the images are classified correctly, and 31.5% are misclassified.

Table 6.3 shows that most errors are done between neighbouring classes. The highest error is due to misclassifications between class 2 and 3.

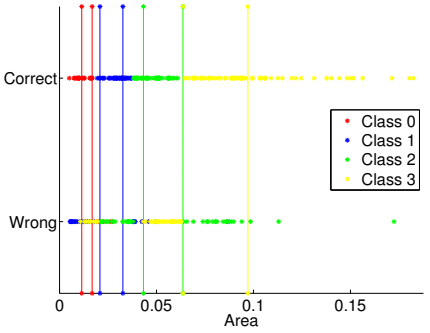


FIGURE 6.4: Classification results for when all images are in both the test set and the training set.

Belongs to class	Classified as class 0	1	2	3
0	22	7	0	0
1	21	46	10	1
2	0	18	53	21
3	7	2	27	61

TABLE 6.3: Overview of misclassifications.

6.2.2.2 Results for different partitions

The database is now partitioned into a training and a test set. As all images are used, increasing the training size will lower the size of the test set. The classification results for different partitions are shown in Table 6.4. These results are found by testing the classifier 100 times for each partition, and then taking the average error.

Training size	Test size	Error
59	237	40.90%
90	206	39.84%
119	177	40.34%
149	147	40.14%
177	119	38.26%
206	90	39.78%
237	59	40.56%

TABLE 6.4: Classification errors for different partitions of images based on the area of inflammation.

These results show that the classification error is close to 40% in all cases. In addition, there is no effect of increasing the training size.

6.2.2.3 Results for fixed test size

The test size will now be fixed to 100 images. We will test the model as the training size is increased from 25 to 170 images. The error for different training

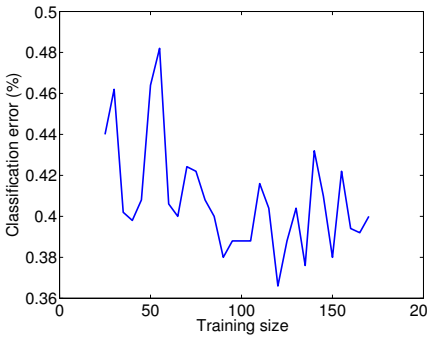


FIGURE 6.5: Classification errors for increasing training size and fixed test size.

sizes is plotted in Figure 6.5. Also here, the error is about 40% in all cases. There is also no effect in increasing the training size. Detailed results may be found in Appendix C.

6.2.3 Comments on the results

Although an error close to 40% is high, the results look rather promising.

Some of the error may be explained by the different sources of error discussed in section 6.1.2. However, even if all these sources were removed, we would still experience misclassifications. There are in general two reasons for this.

First, when an image is classified by a medical doctor, there are only four options. In reality there are many intermediate cases. It might happen that an image should be classified as both degree 1 and 2, because the inflammation is somewhere in between. The medical doctor is however forced to classify it either as degree 1 or degree 2. As we saw in Table 6.3 most errors are done between neighbouring classes which could indicate that there are many intermediate cases.

The second reason is that the class is not only dependent of the area, but also on the shape of the inflamed region. Although it is clear from Figure 6.2(a) that there is a correspondence between the area and the degree of inflammation, the area alone does not seem to tell the whole story. This is also verified by participants in the Medusa project.

The obtained result shows that the area of the inflamed region within each class is normally distributed with roughly the same variance. This can be used as a starting point for developing algorithms by using parameters related for example to the shape of the inflamed region.

6.3 Classification from scattering coefficients

In this section we will present the results from the classification using the scattering coefficients.

6.3.1 Setup

The scattering coefficients may be computed for different values of the parameters J, R and M, which denotes the scale, number of rotations and number of layers

respectively. A dimension d of the affine spaces which approximates the coefficients from the training set must be chosen. As there are four classes, representing different degrees of inflammation, there are in total four affine spaces. These four affine spaces together with the classifier will be referred to as the model. A particular model is thus dependent on the parameters J, R, M, d and the number of images in the training set. To test a model, the scattering coefficients from the test set are used.

A diagram of this procedure is provided in Figure 6.6. The green nodes means there is a parameter which needs to be specified. For instance, how many images should be in the training/test set, and what is the dimension of the affine spaces. As we see, there are a lot of combinations to be investigated.

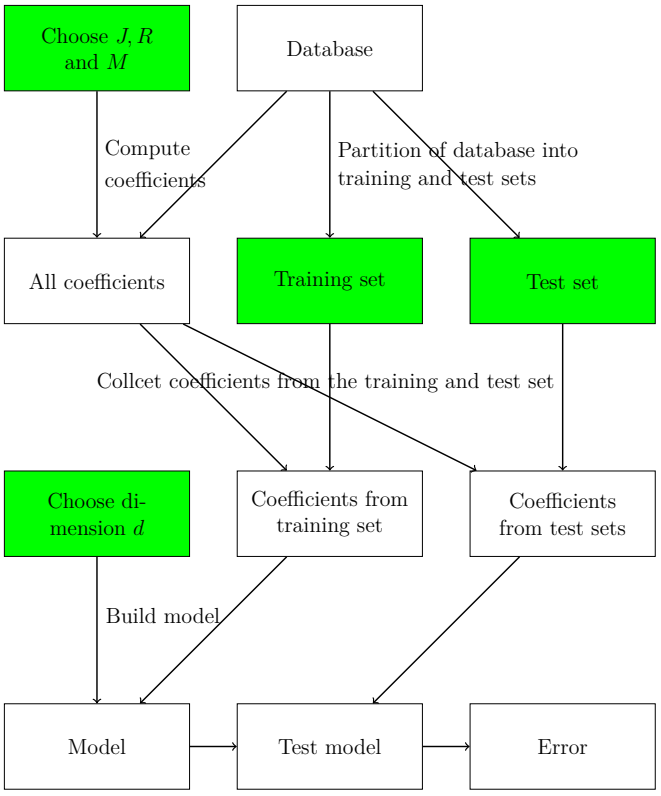


FIGURE 6.6: Diagram of classification algorithm from scattering coefficients. Nodes with green color means that there is a choice to be made.

The scattering coefficients are computed at scales $J = 3, 4, \cdots, 8$ with $M = 3$. As the images are of size $256 \times 256 = 2^8 \times 2^8$, the largest possible pixel displacement is of the order 2^8. It is therefore not necessary to include scales with $J > 8$. For each scale the wavelet is rotated by $R = 2, 3, \cdots, 8$ number of rotations. Hence there are in total 42 different sets of coefficients. Further, we will test each model with $d = 0, 1, 2, \cdots, 30$. Since the smallest class of images contains 29 images, we would expect overfitting if the dimension is further increased. Hence for each partition of training and test sets there are $42 \times 30 = 1260$ different outputs.

	Degree 0	Degree 1	Degree 2	Degree 3	Total
Train/Test	12/17	31/47	37/55	39/58	119/177
Train/Test	15/14	39/39	46/46	49/48	149/147
Train/Test	17/12	47/31	55/37	58/39	177/119
Train/Test	20/9	55/23	64/28	68/29	207/89

TABLE 6.5: Number of images with different degree of inflammation in each partition of training and test sets.

6.3.2 Results from scattering classifier

The results from the classification based on the scattering coefficients will now be presented. Error plots for different partitions of the database and a discussion of optimal parameters will be given. Detailed results may be found in Appendix B.

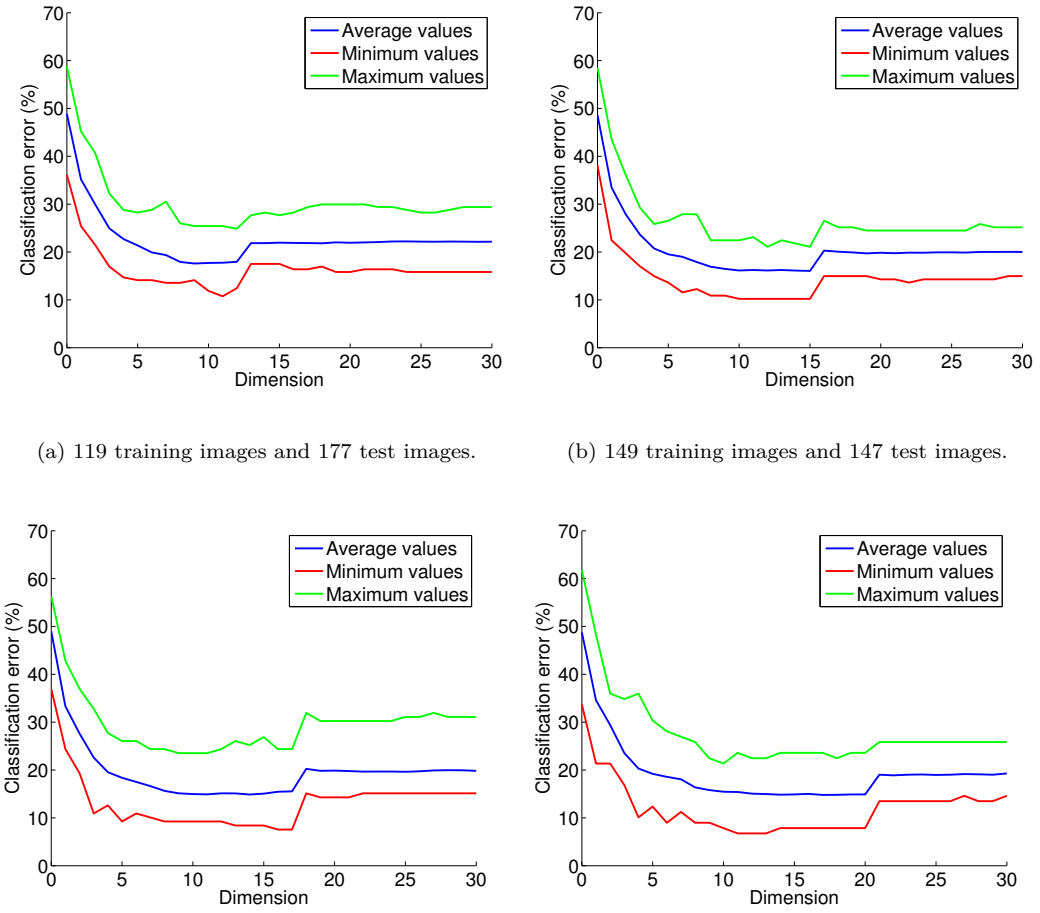
6.3.2.1 Results for different partitions of the database

The whole database is partitioned into a training set, and a test set. The different partitions are found in Table 6.5.

For each partition the error is computed for all possible combination of J, R and d. A plot of the error as function of the dimension d for each partition is illustrated in Figure 6.7. For each dimension d there are in total 42 different outputs, corresponding to the different combinations of J and R. The red line is the minimum error of all these combinations, the blue line is the average error and the green line is the maximum error. All these plots have a similar shape. For low dimensions ($d < 8$), the error is large since much of the variability is not captured by the affine space approximation, i.e underfitting. As the dimension increases from $d = 0$ to $d = 8$ there is a drop in the error from about 45% to 20% in the average case. Increasing the dimension further will give about the same error as in the case when $d = 8$. At a particular point, which is dependent on the partition, there is a rapid increase in the error. This is the effect of overfitting, which means that random error becomes prominent in the model. Overfitting will happen if the dimension d exceeds the number of training images in the smallest class. In Table 6.5 we see the number of training images within each class for the different partitions. The smallest class corresponds to images with inflammation degree 0. We would therefore expect overfitting for the partition with 119 training images when d exceeds 12. This can be seen from Figure 6.7(a). After this increase, the error appears to remain approximately constant for higher dimensions.

It is also interesting to investigate which errors that are made by our classifier. In Table 6.6 to 6.9, we present the percentage of all classifications for each class. As we can see, most misclassifications are due to images of degree 0 which are classified as degree 1. The reason for this is because images with degree 0 have the smallest number of training images. From a medical point of view, this is not a crucial error. It would for instance be much worse if many images of degree 3 were classified as degree 0.

The minimum error achieved, and the corresponding dimension of the affine spaces is provided in Table 6.10. These errors are also seen as the minimum values



(a) 119 training images and 177 test images.

(b) 149 training images and 147 test images.

(c) 177 training images and 119 test images.

(d) 206 training images and 90 test images

FIGURE 6.7: Classification error as a function of the dimension of the affine spaces.

Class	Classified as class			
	0	1	2	3
0	71.01	24.23	2.66	2.10
1	7.6494	84.3972	5.6738	2.28
2	0.95	6.49	80.95	11.61
3	0.21	1.52	9.77	88.51

TABLE 6.6: Percentage of classifications for each class with 119 training images and 177 test images.

Class	Classified as class			
	0	1	2	3
0	79.08	15.99	3.23	1.70
1	7.63	85.90	4.52	1.95
2	0.47	4.71	85.51	9.32
3	0.05	0.84	9.97	89.14

TABLE 6.7: Percentage of classifications for each class with 149 training images and 147 test images.

in Figure 6.7. This shows that by increasing the training size, and at the same time decreasing the test size, one will obtain a lower minimum error. This is not surprising. Next, we will see the results when the test size is fixed, and the training size is varied.

Class	Classified as class			
	0	1	2	3
0	79.56	17.66	2.18	0.60
1	8.30	86.18	3.76	1.77
2	0.90	4.89	87.00	7.21
3	0.31	0.86	8.85	89.99

TABLE 6.8: Percentage of classifications for each class with 177 training images and 119 test images.

Class	Classified as class			
	0	1	2	3
0	80.76	15.45	2.44	1.36
1	7.85	87.91	2.65	1.59
2	1.13	4.18	88.42	6.27
3	0.00	0.93	7.99	91.09

TABLE 6.9: Percentage of classifications for each class with 206 training images and 90 test images.

Training / Test size	Minimum error	Optimal dimension
119 / 177	10.73 %	11
149 / 147	10.2 %	10
177 / 119	7.56 %	16
206 / 90	6.74 %	11

TABLE 6.10: Minimum errors and the corresponding dimension for different partitions of training and test sets.

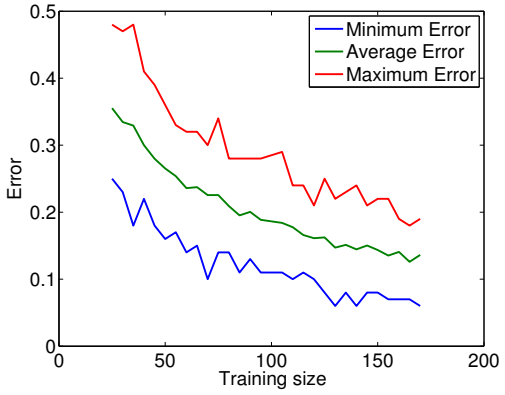


FIGURE 6.8: Classification error as a function of training size. The error is computed with the optimal dimension of the affine spaces. The different lines shows the maximum, average and minimum error achieved with different combinations of J and R.

6.3.2.2 Results for fixed test size

We would like to see the effect of increasing the training size, when the test size is kept fixed. Hence the test size will now be fixed to 100 images. In Figure 6.8 the error is plotted as a function of the training size. For each training size there is a choice for the parameters d, J and R. In this plot, the dimension d which produces the minimum error is chosen. Hence, the minimum, average and maximum error lines, are computed over different combinations of J and R. It is clear from this figure that increasing the test size will decrease the classification error.

In Figure 6.9 the dimension with the smallest error for each training size is plotted. This dimension is also the dimension used to calculate the error in Figure 6.8. This indicates that when the training size is increasing, then so is the

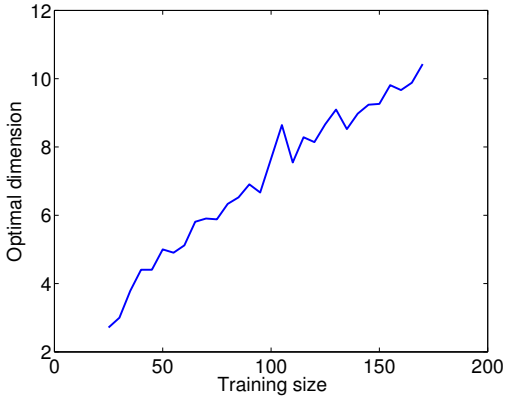
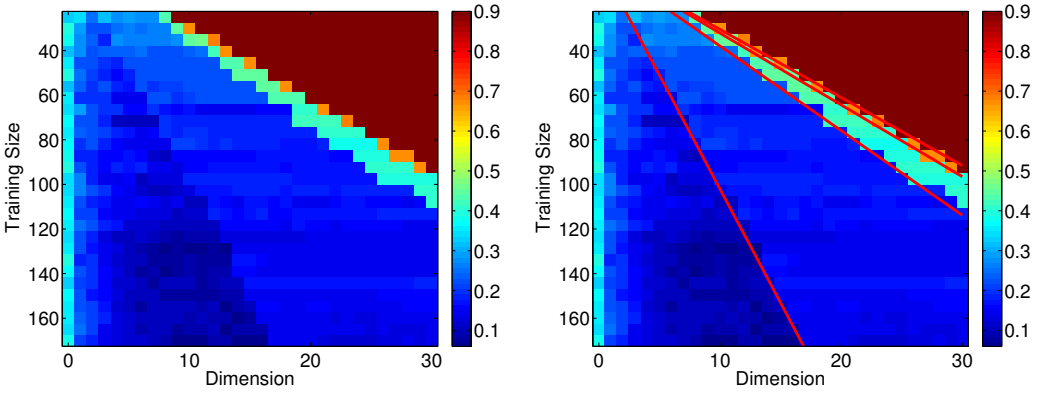


FIGURE 6.9: Dimension for which minimum error is achieved for different training sizes.



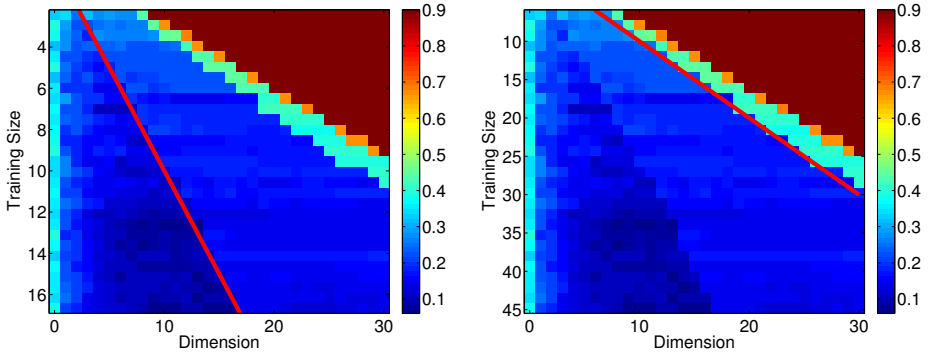
(a) (b)

FIGURE 6.10: Training size against dimension.

dimension of the affine spaces which yields the minimum error.

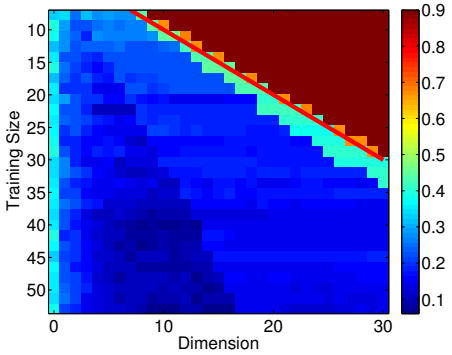
The error for different combinations of training sizes and dimensions of the affine spaces is visualized in Figure 6.10(a). This figure illustrates an important observation. Notice the increase in the error across the lines plotted in Figure 6.10(b). The dimension plotted against the training size for each class, is visualized in Figure 6.11. The red line shows where the training size, and the dimension coincide. As the dimension increases it will exceed the number of training images in the different classes. When it does, there is an increase in the error due to overfitting.

The error is plotted against the dimension of the affine spaces in Figure 6.12. The minimum error for each dimension follows about the same curve as the error for the maximum training size tested. In the average case, the minimum error is attained when the dimension is close to 8. For smaller dimensions, the model will suffer from underfitting. Higher dimensions will give a lower error if the training size is properly adjusted, but in the average case the error is affected by overfitting from the sets with small training sizes.

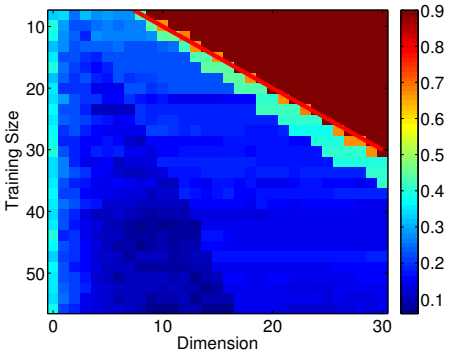


(a) Class 0.

(b) Class 1.



(c) Class 2.



(d) Class 3.

FIGURE 6.11: Training size for each class against dimension.

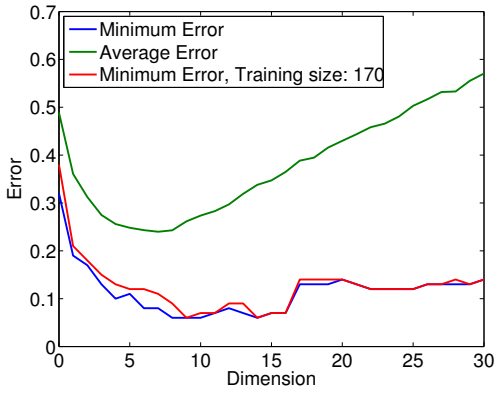
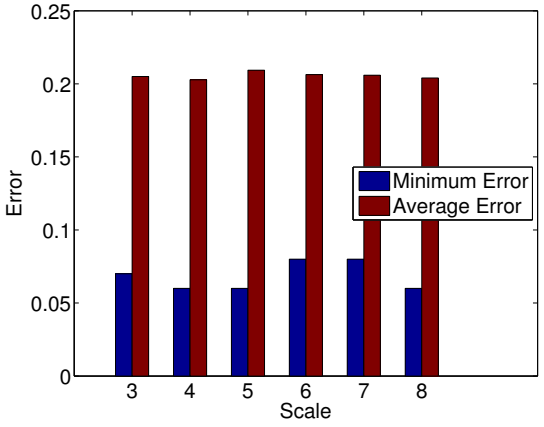
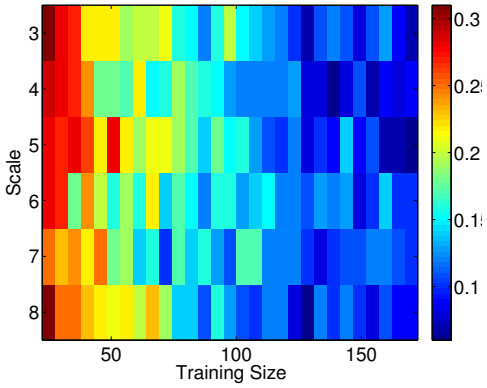


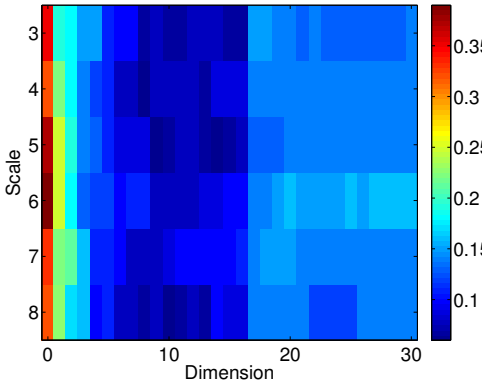
FIGURE 6.12: Classification error as a function of dimension of affine spaces.



(a) Error for different scales.



(b) Training size agains scale.



(c) Training size agains number of rotations.

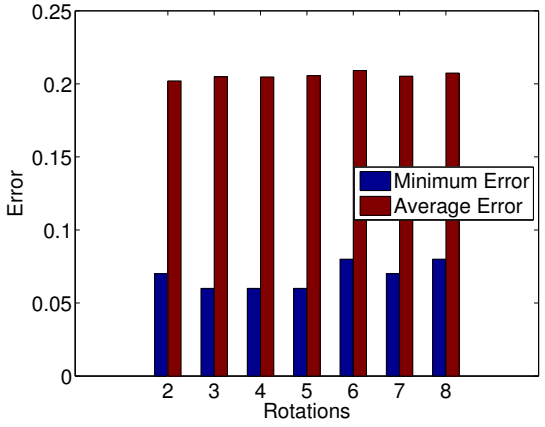
FIGURE 6.13: Error plots for the scale J.

6.3.2.3 Optimal scale and number of rotations

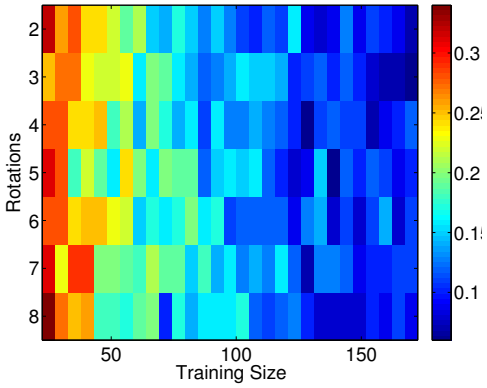
We would like to find the optimal scale and number of rotations, on the base of the above numerical results. By optimal we mean the value of these parameters which gives the smallest classification error.

We will first look at the scale. In Figure 6.13, the error is plotted for the different choices of the scale variable. Figure 6.13(a) shows a histogram of the average and minimum error attained for different scales. The error is computed with the optimal dimension, and the average is taken over different combinations of training sizes and number of rotations. There are only minor differences between the different choices. Therefore, one may wonder if there is a dependence between the scale and the other parameters.

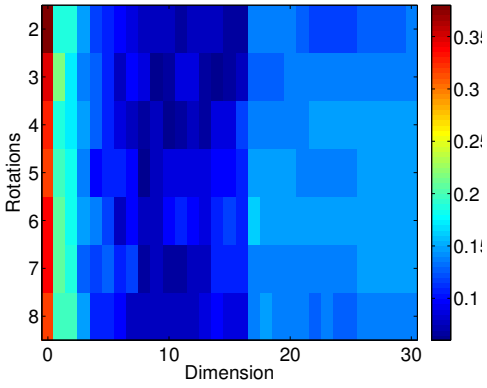
In Figure 6.13(b) and 6.13(c) the minimum error is plotted for different scales against the training size and dimension of the affine spaces respectively. These figures seem to indicate that small scales ($J = 3, 4, 5$) produce the smallest classification error. However, the results are not significant.



(a) Error for different number of rotations.



(b) Training size agains scale.



(c) Training size agains number of rotations.

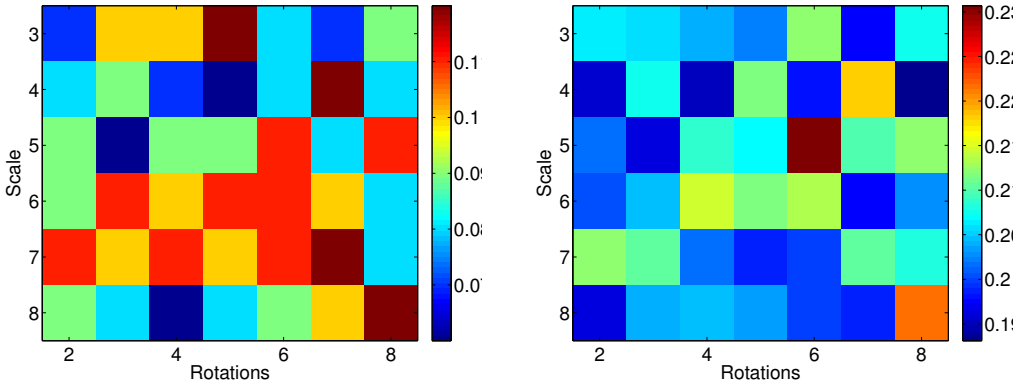
FIGURE 6.14: Error plots for the number of rotations R.

We will now consider the number of rotations. In Figure 6.14 the error is illustrated for different choices of number of rotations. A few number of rotations ($R \leq 5$) seem to give the smallest error, but also here the differences are small.

It might happen that it is the combination of the scale and number of rotations which is important. In Figure 6.15, the minimum and average error is visualized for different combinations of scales and number of rotations. The average is taken over all training sets where the optimal dimension is chosen for each set.

There are some combinations which produce a smaller error than others. The minimum error of 6% is attained for the combinations $(J, R) = (4, 5), (5, 3), (8, 4)$. The minimum average error of 19.3% is attained for $(J, R) = (4, 8)$.

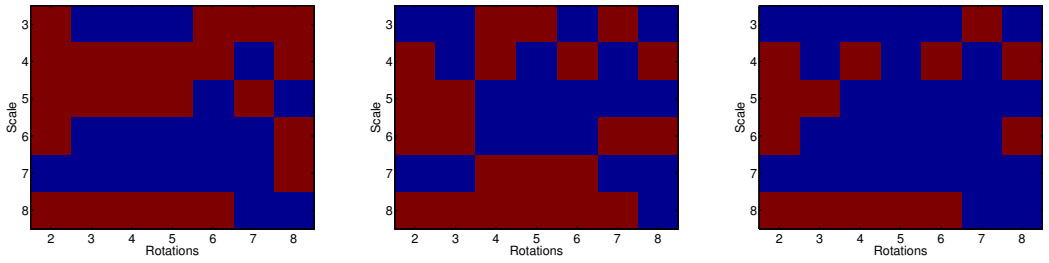
The average minimum error is about 9%, whilst the overall average error is about 20.5%. In Figure 6.16(a), the combinations which give a minimum error smaller than the average minimum error is displayed as red. Similarly, the combinations which give an average error smaller than the overall average error is displayed in Figure 6.16(b). The intersection of these two figures is illustrated in Figure 6.16(c).



(a) Minimum error.

(b) Average error.

FIGURE 6.15: Error for different combinations of scales and number of rotations.



(a) Minimum error less than average minimum error

(b) Average error less than overall average error

(c) Intersection of 6.16(a) and 6.16(b)

FIGURE 6.16: Illustrates which combinations of J and R which gives the smallest error.

From these Figures we see that with $J = 4$, a small error is attained in both the average and minimum case when the number of rotations is even. In addition $J = 8$ seems to give a small error when the number of rotations are less than 7. Still these results are not significant enough to conclude that some combinations are more optimal than others. The recommendation is therefore to choose the combination which gives the minimum computational cost. In this case, this would be the combination $(J, R) = (8, 2)$.

6.4 Summary of the results

6.4.1 Area of inflammation

The results show that there is a classification error of 40%, and increasing the training size does not seem to lower the error. The high error is partly explained by the sources of error, but even with all these sources removed we would still experience a high number of misclassifications. Most misclassifications are done

between neighbouring classes. This indicates that there are many intermediate cases, for which a more thorough examination should be conducted. A class is not alone determined by the area, but also on the shape of the inflamed region.

6.4.2 Scattering coefficients

The average error obtained from the classification algorithm using the scattering coefficients is about 20%. In the best cases the error is close to 6%.

Increasing the training size shows in general a decrease in the error. The dimension of the affine spaces should however be adjusted to the number of images in the class with the least amount of images. If the dimension exceeds the amount of images in this class, one will experience overfitting, and thus an increase in the number of misclassifications. The optimal dimension of these affine spaces should therefore be chosen as high as possible, but smaller than the number of training images in the smallest class.

The scale variable J controls the amount of translation invariance. A higher value for J implies that the coefficients are more translation invariant, but also that the scattering distance between any two images is smaller. The results show small differences in the choice of J. This could indicate that the variability within each class is not primarily due to translation, but to deformations and additive noise.

The same goes to the number of rotations R. No particular choice of this parameter shows a significant decrease in the error.

The recommendation is therefore to choose a high scale and a small number of rotations so that the computational expenses, and memory usage are minimal.

Chapter 7

Conclusion

In this thesis we have studied and applied the theory and techniques of the windowed scattering transform to the problem of image classification. As a model case we considered US images of finger-joints. This database of images is a part of a Norwegian-Polish research project called Medusa. The goal of the Medusa project is to develop a software for classification and recognition of synovitis. Synovitis is a type of inflammation occurring in finger-joints.

The method proved to be rather efficient. The results show that the amount of training images are important, and that the dimension of the affine space approximation should be adjusted to the training size.

A pure analytic analysis of this method has been conducted. We have proved that the scattering metric is locally translation invariant, stable to additive noise and stable to the action of diffeomorphisms. In order to apply the theory we have analyzed the general pattern and adjusted it to our concrete setting. In particular error estimates for specific choice of wavelets, and a new proof showing that the energy is captured by frequency-decreasing paths is given. The corresponding numerical algorithms have been developed in order to obtain the scattering coefficients. The latter have been classified by considering projections of affine spaces which are constructed using principal component analysis. This is a statistical method used for dimensionality reduction.

Other methods for classification have also been studied, in particular a method based on the area of inflammation. The results from this method show that it is not very effective. This is due to the high variation in the area within each class, and that the degree of inflammation is not only determined by the area but also by the shape of the inflamed region.

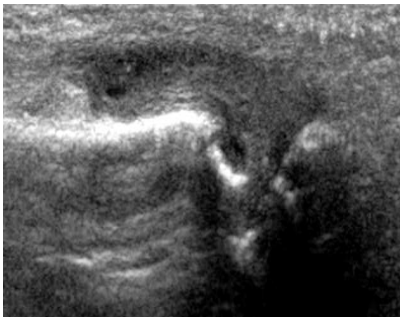


FIGURE 7.1: Typical US image of synovitis.

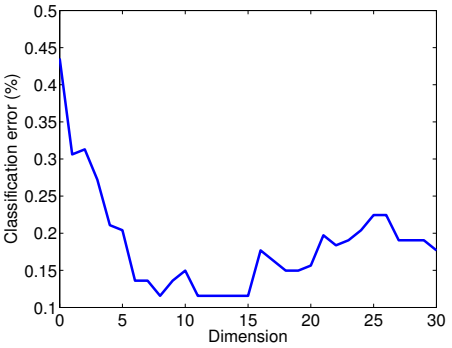


FIGURE 7.2: Classification error for different dimensions.

A typical US image of synovitis is shown in Figure 7.1. Scattering coefficients are computed by convolving the image with a rotated and dilated wavelet of the form

$$\psi(x) = e^{-\frac{|x|^2}{2\sigma^2}} \left(e^{i\langle x,\xi \rangle} - e^{-(\sigma^2|\xi|^2)/2} \right).$$

Thereafter, a non-linear modulus operator is applied. The wavelet may be rotated by two rotations, corresponding to 0 and $\frac{\pi}{2}$ radians. Further it may be dilated by three dilations, corresponding to the scales $2^0, 2^1$ and 2^2. This convolution- and modulus procedure is then repeated with different combinations of dilations and rotations, typically three times. The number of such repetitions is referred to as the number of layers. The results after cascading wavelet transforms with a non-linear modulus are filtered with a low-pass filter of the form

$$\phi(x) = \frac{1}{2\pi\sigma} e^{-\frac{|x|^2}{2\sigma}},$$

scaled according to the coarsest scale. With this choice of scales for the wavelet, the coarsest scale would be 2^3. The output for a particular image, will be a vector of scattering coefficients corresponding to the different combinations of scales and rotations. In addition to scales, rotations and number of layers, the length of this vector depends on the size of the input image, and the coarsest scale. The images considered in this thesis are of size 256×256 pixels. This procedure is then applied to all images in the available database, which in our case contains 296 images. All images belong to one of four classes depending on the degree of inflammation. To classify images, the database is divided into a training set and a test set. Images from the training set are used to build affine spaces, which approximate the scattering coefficients for each class. These spaces are constructed using principal component analysis. A dimension of these affine spaces must also be chosen. Images from the test set are then classified according to projections on these affine spaces. The error is measured by the ratio between number of misclassifications and the number of images in the test set. In Figure 7.2 we see the error plotted against the dimension of the affine spaces, for the case with two rotations, three scales and three layers. Here the database has been divided evenly between the training set and the test set.

7.1 Suggestions for further work

Imposing more invariant structure on the scattering coefficients. The scattering coefficients used in this thesis are proven to be invariant to translations. There exists however an extension of these coefficients which are invariant under the action of any compact Lie group. Examples are coefficients which are both rotation and translation invariant. Classification based on these coefficients has proven to be very efficient in texture classification, and it would be interesting to see the results for classification of synovitis.

Combined scattering and edge-detection. We saw in section 5.2.1 that finding the correct edges which defines the boundary of the synovitis is difficult. The degree of the inflammation is determined by the area as well as the shape of the inflammation. With prior knowledge of the degree of inflammation, one can search for specific shapes of the inflammation. This may be done by first classifying images based on their scattering coefficients, and then adjusting the edge-detector by favouring edges which define shapes corresponding to the classification result. In this way, one can develop a method which both classifies, and locates the boundary of the inflammation.

Apply techniques to other types of US images. In this thesis we study a very special class of images. After appropriate adjustments, the method can be applied to much more general classes. The work done in this thesis should therefore serve as a model case for a wider class of images. Today US images are used in a broad range of medical disciplines. It would therefore be interesting to see these techniques applied to other types of US images.

Contribution to the Medusa project. The work done in this thesis is a direct contribution to the Medusa project. With proper adjustments, the classification method based on the scattering coefficients may easily be implemented as a part of a more extensive algorithm for classification and recognition of synovitis.

Appendix A

Some results from Mathematical Analysis

Theorem A.1. *(**Schur Test**, see e.g [Grö01, p.106]) Let $K(x,y)$ be a measurable function on $\mathbb{R}^2$ that satisfies the conditions*

$$\sup_{x\in\mathbb{R}}\int_{\mathbb{R}}|K(x,y)|dy\leq K_1,\quad \textit{and}\quad \sup_{y\in\mathbb{R}}\int_{\mathbb{R}}|K(x,y)|dx\leq K_2. \tag{A.1}$$

Then the integral operator A defined by $Af(x)=\int_{\mathbb{R}}K(x,y)f(y)dy$ is bounded from $L^2(\mathbb{R})$ to $L^2(\mathbb{R})$ and

$$\|A\|_{L^2\to L^2}\leq\sqrt{K_1K_2}. \tag{A.2}$$

Theorem A.2. *(**Tonelli's Theorem** see e.g [MW13, p.212]) Suppose $(\Gamma,\mathcal{S},\mu)$ and $(\Lambda,\mathcal{T},\nu)$ are σ-finite measure spaces. Let f be a nonnegative extended real-valued $\mathcal{S}\times\mathcal{T}$-measurable function on $\Gamma\times\Lambda$. Then*

$$\int_{\Gamma\times\Lambda}f(x,y)d(\mu\times\nu)(x,y)=\int_{\Gamma}\left[\int_{\Lambda}f(x,y)d\nu(y)\right]d\mu(x) \tag{A.3}$$

$$=\int_{\Lambda}\left[\int_{\Gamma}f(x,y)d\mu(x)\right]d\nu(y). \tag{A.4}$$

Definition A.3. *(**1D Fourier Transform**) For a function $f\in L^2(\mathbb{R})$ the Fourier transform is*

$$\widehat{f}(\omega)=\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty}f(t)e^{-i\omega t}dt. \tag{A.5}$$

Definition A.4. *(**2D Fourier Transform**) For a function $f\in L^2(\mathbb{R}^2)$ the Fourier transform is*

$$\widehat{f}(\omega)=\frac{1}{2\pi}\int_{\mathbb{R}^2}f(t)e^{-i\langle\omega,t\rangle}dt. \tag{A.6}$$

Here $t=(t_1,t_2)$ and $\omega=(\omega_1,\omega_2)$.

Definition A.5. *(**Convolution**) Let $f, g \in L^2(\mathbb{R}^2)$. The convolution is defined almost everywhere as*

$$f * g(x) = \int_{\mathbb{R}^2} f(y)g(x-y)dy. \tag{A.7}$$

Theorem A.6. *([GW99, Prop 23.2.1]) If $f, g \in L^2(\mathbb{R}^2)$, then*

$$\widehat{f * g}(\omega) = \widehat{f}(\omega) \cdot \widehat{g}(\omega) \tag{A.8}$$

for all $\omega \in \mathbb{R}^2$.

Proposition A.7. *([GW99, Prop 20.2.1]) If $f, g \in L^1(\mathbb{R}^n)$, then $f * g$ exists almost everywhere and $f * g$ belongs to $L^1(\mathbb{R}^n)$. Moreover*

$$\|f * g\|_{L^1} \leq \|f\|_{L^1}\|g\|_{L^1}. \tag{A.9}$$

Proposition A.8. *([GW99, Prop 20.3.2]) If $f \in L^1(\mathbb{R}^n)$ and $g \in L^2(\mathbb{R}^n)$, then the following hold:*

- *$f * g$ exists almost everywhere.*
- *$f * g$ belongs to $L^2(\mathbb{R}^n)$ and*

$$\|f * g\|_{L^2} \leq \|f\|_{L^1}\|g\|_{L^2}. \tag{A.10}$$

Proposition A.9. *([GW99, Prop 21.2.1]) Let f be in $L^1(\mathbb{R})$ and let g be in $C^n(\mathbb{R})$. Assume that $g^{(k)}$ is bounded for $k = 0, 1, \cdots, n$. Then*

- *$f * g \in C^n(\mathbb{R})$, and*
- *$(f * g)^{(k)} = f * g^{(k)}$ for $k = 0, 1, \cdots, n$.*

Theorem A.10. *(**The Plancherel equality** see e.g [GW99, Thm 22.1.2])For all $f, g \in L^2(\mathbb{R})$,*

$$\int_{\mathbb{R}} f(x)g^*(x)dx = \int_{\mathbb{R}} \widehat{f}(\omega)\widehat{g}^*(\omega)d\omega. \tag{A.11}$$

Theorem A.11. *(**The Parseval equality** see e.g [GW99, Thm 22.1.2])For all $f \in L^2(\mathbb{R})$,*

$$\int_{\mathbb{R}} |f(x)|^2 dx = \int_{\mathbb{R}} |\widehat{f}(\omega)|^2 d\omega. \tag{A.12}$$

Appendix B

Classification results from scattering coefficients

TABLE B.1: Classification results with $J = 3$ and $M = 3$.

R	N_J	Train/Test size = 119/177		Train/Test size = 149/147		Train/Test = 177/119		Train/Test size = 206/90	
		Min error(%)	Opt. dim	Min error(%)	Opt. dim	Min error(%)	Opt. dim	Min error(%)	Opt. dim
2	27648	13.5593	8	12.9252	10	8.4034	13	16.8539	10
3	65536	16.9492	7	10.8844	11	9.2437	5	7.8652	12
4	128000	18.6441	10	17.0068	13	21.0084	8	14.6067	19
5	221184	22.0339	10	12.9252	15	13.4454	16	12.3596	16
6	351232	14.6893	9	14.966	13	12.605	12	10.1124	19
7	524288	19.209	8	14.2857	9	10.9244	14	11.236	13
8	746496	18.0791	10	16.3265	7	15.1261	15	16.8539	12

TABLE B.2: Classification results with $J = 4$ and $M = 3$.

R	N_J	Train/Test size = 119/177		Train/Test size = 149/147		Train/Test = 177/119		Train/Test size = 206/90	
		Min error(%)	Opt. dim	Min error(%)	Opt. dim	Min error(%)	Opt. dim	Min error(%)	Opt. dim
2	16640	11.8644	11	12.9252	7	11.7647	10	6.7416	11
3	44800	14.6893	11	17.6871	6	14.2857	5	10.1124	10
4	94464	15.2542	7	12.9252	7	8.4034	13	8.9888	10
5	171776	13.5593	7	12.9252	11	11.7647	8	8.9888	11
6	282880	12.9944	12	12.9252	15	11.7647	11	13.4831	13
7	433920	16.3842	9	11.5646	12	15.9664	14	10.1124	10
8	631040	17.5141	8	14.966	5	15.1261	8	16.8539	6

TABLE B.3: Classification results with $J = 5$ and $M = 3$.

R	N_J	Train/Test size = 119/177		Train/Test size = 149/147		Train/Test = 177/119		Train/Test size = 206/90	
		Min error(%)	Opt. dim	Min error(%)	Opt. dim	Min error(%)	Opt. dim	Min error(%)	Opt. dim
2	8384	14.6893	9	13.6054	5	11.7647	12	13.4831	9
3	24064	15.8192	9	18.3673	10	10.084	11	14.6067	5
4	52544	15.8192	9	17.0068	8	9.2437	11	14.6067	13
5	97664	16.9492	9	16.3265	11	10.9244	3	14.6067	18
6	163264	14.1243	11	15.6463	10	14.2857	12	15.7303	7
7	253184	14.6893	12	19.0476	9	15.9664	14	13.4831	13
8	371264	16.3842	8	19.0476	13	10.9244	9	12.3596	9

TABLE B.4: Classification results with $J = 6$ and $M = 3$.

R	N_J	Train/Test size = 119/177		Train/Test size = 149/147		Train/Test = 177/119		Train/Test size = 206/90	
		Min error(%)	Opt. dim	Min error(%)	Opt. dim	Min error(%)	Opt. dim	Min error(%)	Opt. dim
2	3728	14.6893	7	11.5646	9	10.9244	9	7.8652	16
3	11104	20.339	10	16.3265	14	11.7647	11	10.1124	10
4	24720	14.1243	12	14.966	7	10.9244	14	13.4831	18
5	46496	16.9492	8	12.2449	12	10.084	8	13.4831	8
6	78352	14.1243	9	15.6463	14	10.9244	15	12.3596	19
7	122208	18.6441	7	14.966	10	14.2857	6	16.8539	19
8	179984	18.0791	8	11.5646	9	7.563	16	13.4831	9

TABLE B.5: Classification results with $J = 7$ and $M = 3$.

R	N_J	Train/Test size = 119/177		Train/Test size = 149/147		Train/Test = 177/119		Train/Test size = 206/90	
		Min error(%)	Opt. dim	Min error(%)	Opt. dim	Min error(%)	Opt. dim	Min error(%)	Opt. dim
2	1516	18.0791	8	14.2857	14	18.4874	11	15.7303	9
3	4624	19.774	10	16.3265	10	9.2437	11	13.4831	12
4	10420	18.6441	5	18.3673	9	15.1261	17	13.4831	11
5	19744	17.5141	7	10.2041	12	13.4454	5	10.1124	11
6	33436	13.5593	10	10.2041	10	15.9664	4	10.1124	8
7	52336	20.339	11	14.2857	15	23.5294	6	12.3596	17
8	77284	14.6893	9	16.3265	6	10.9244	7	12.3596	7

TABLE B.6: Classification results with $J = 8$ and $M = 3$.

R	N_J	Train/Test size = 119/177		Train/Test size = 149/147		Train/Test = 177/119		Train/Test size = 206/90	
		Min error(%)	Opt. dim	Min error(%)	Opt. dim	Min error(%)	Opt. dim	Min error(%)	Opt. dim
2	577	18.0791	7	16.3265	8	15.1261	14	11.236	17
3	1789	12.4294	12	13.6054	7	12.605	10	14.6067	10
4	4065	10.7345	11	14.966	9	12.605	13	15.7303	12
5	7741	24.8588	5	15.6463	12	16.8067	8	12.3596	11
6	13153	16.3842	7	14.2857	8	10.084	9	21.3483	9
7	20637	15.2542	12	12.2449	6	15.1261	5	15.7303	11
8	30529	14.1243	6	16.3265	15	15.1261	13	11.236	20

Training size	Error(%)	Scale J	Rotation R	Dimension d
25	25	2	7	3
30	27	8	5	8
35	18	3	6	5
40	22	7	3	5
45	18	4	4	8
50	16	4	6	5
55	17	5	4	8
60	14	5	7	4
65	15	11	4	2
70	10	6	7	8
75	14	6	8	4
80	14	5	7	3
85	11	8	8	4
90	13	9	7	3
95	11	6	7	6
105	11	7	8	2
110	10	10	8	2
115	11	9	5	8
120	10	8	5	5
125	8	9	8	5
130	6	10	8	4
135	8	8	4	2
140	6	8	4	5
145	8	7	4	8
150	8	8	3	6
155	7	13	4	4
160	7	14	5	3
165	7	13	5	3
170	6	9	5	3

TABLE B.7: Minimum error for different training sizes, and the corresponding optimal parameters.

Training size	Min error(%)	Mean error(%)	Max error(%)	Mean dimension
25	25	35.5238	48	2.7143
30	27	34.6667	48	3
35	18	32.9286	48	3.7857
40	22	30	41	4.4048
45	18	27.9762	39	4.4048
50	16	26.5238	36	5
55	17	25.381	33	4.9048
60	14	23.5952	32	5.119
65	15	23.7381	32	5.8095
70	10	22.5476	30	5.9048
75	14	22.5476	34	5.881
80	14	20.9048	28	6.3333
85	11	19.5238	28	6.5238
90	13	20.0476	28	6.9048
95	11	18.8571	28	6.6667
105	11	18.4048	29	8.6429
110	10	17.7619	24	7.5476
115	11	16.5952	24	8.2857
120	10	16.119	21	8.1429
125	8	16.2381	25	8.6667
130	6	14.7143	22	9.0952
135	8	15.119	23	8.5238
140	6	14.4524	24	8.9762
145	8	15.0238	21	9.2381
150	8	14.3571	22	9.2619
155	7	13.5238	22	9.8095
160	7	14.0714	19	9.6667
165	7	12.5952	18	9.881
170	6	13.619	19	10.4286

TABLE B.8: Minimum and average error for increasing training sizes, and the average optimal dimension.

Appendix C

Classification results from area of inflammation

Training size	Error(%)
25	44.0
30	46.2
35	40.2
40	39.8
45	40.8
50	46.4
55	48.2
60	40.6
65	40.0
70	42.4
75	42.2
80	40.8
85	40.0
90	38.0
95	38.8
105	38.8
110	41.6
115	40.4
120	36.6
125	38.8
130	40.4
135	37.6
140	43.2
145	41.0
150	38.0
155	42.2
160	39.4
165	39.2
170	40.0

TABLE C.1: Error as the function of training size for fixed test size.

Bibliography

[AMVA04] J. Antoine, R. Murenzi, P. Vandergheynst, and S. Ali. *Two-Dimensional Wavelets and their Relatives*. Cambridge University Press, 2004.

[ASM+14] J. Andén, L. Sifre, S. Mallat, M. Kapoko, V. Lostanlen, and E. Oyallon. Scatnet. http://www.di.ens.fr/data/software/scatnet/, January 2014.

[BM12] J. Bruna and S. Mallat. Invariant scattering convolution networks. *IEEE Transactions on pattern analysis and machine intellegence*, 35(8):1872–1886, March 2012.

[Bru12] J. Bruna. *Scattering Represenations for Recognition*. PhD thesis, Ecole Polytechnique, November 2012.

[Can86] J. Canny. A computational approach to edge detection. *IEEE Transactions on pattern analysis and machine intellegence*, PAMI-8(6):679–698, November 1986.

[Dau85] J. Daugman. Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters. *Optical Society of America, Journal, A: Optics and Image Science*, 2(7):1160–1169, July 1985.

[EC12] B. K. Ersbøll and K. Conradsen. *Multivariate Statistics - An Introduction*. DTU Informatics, Department of Informatics and Mathematical Modeling, 8 edition, 2012.

[Grö01] K. Gröchenig. *Foundations of Time-Frequency Analysis*. Birkhäuser, 2001.

[GW99] C. Gasquet and P. Witomski. *Fourier Analysis and Applications*. Springer, 1999.

[GW08] R. Gonzalez and R Woods. *Digital image processing*. Upper Saddle River, N.J. : Pearson/Prentice Hall, 3 edition, 2008.

[KSJS00] E. Kansdel, J. Schwartz, T. Jessel, and S. Hudspeth A. Siegelbaum. *Principles of Neural Science*. The McGraw-Hill Companies, 4 edition, 2000.

[Mal09] S. Mallat. *A wavelet tour of signal processing: the sparse way*. Elsevier/Academic Press, 3 edition, 2009.

[Mal12] S. Mallat. Group invariant scattering. *Communications on Pure and Applied Mathematics*, 65(10):1331–1398, October 2012.

[MW13] J. McDonald and N. Weiss. *A course in Real Analysis*. Elsevier/Academic Press, 2 edition, 2013.

[PMA+12] T. Poggio, J. Mutch, F. Anselmi, L. Rosasco, J. Leibo, and A. Tacchetti. The computational magic of the ventral stream: sketch of a theory (and why some deep architectures work). *MIT-CSAIL-TR-2012-035*, December 2012.

[WdM13] Irène Waldspurger, Alexandre d'Aspremont, and Stéphane Mallat. Phase recovery, maxcut and complex semidefinite programming. July 2013.